10079-00A0
APRIL 1973

NAKED MINI LSI/ALPHA LSI
I/O INTERFACE DESIGN GUIDE

COMPUTER AUTOMATION
the NAKED MINI company

CAI Limited,
95a High Street,
Rickmansworth, Herts, WD3 1RB
Tel: Rickmansworth 79704/7 Telex: 922654

COMPUTER AUTOMATION, INC.
the NAKED MINI company
18651 Von Karman, Irvine, Calif. 92664
tel. 714-833-8830 TWX 910-595-1767

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

TABLE OF CONTENTS (Cont'd)

LIST OF ILLUSTRATIONS

LIST OF TABLES

Section 1
GENERAL INFORMATION

1.1 INTRODUCTION

The ALPHA 16 LSI and NAKED MINI 16 LSI computers are highly flexible system components designed to be easily applied to control, communications, and monitoring tasks. These computers are extremely easy to program using assembly or machine language. Organization of the Processor enables the computer to obtain high memory efficiency, avoiding the problem of "core burning", so prevalent in many computers. Memory utilization is further enhanced by the powerful and flexible I/O command set. The I/O structure is simple and efficient, sharply reducing the amount of I/O logic required by units interfacing to the Processor.

To maximize the benefits of the Processor's unique I/O structure, careful consideration must be given to the design of the peripheral controller interfaces. This document presents comprehensive discussions of I/O (1) capability, (2) characteristics, (3) timing, (4) interrupts, (5) electrical design and (6) mechanical design. The discussions that follow refer to the ALPHA 16 LSI and NAKED MINI 16 LSI computers jointly as the ALPHA LSI computer.

1.2 CONTROL MODES

Two types of I/O instructions, select and sense, provide control information to and from an interface. The select instructions establish operating modes, control interrupts or initialize the interface. The sense instructions permit the Processor to obtain the operational status of an interface.

1.3 INPUT/OUTPUT MODES

The ALPHA LSI computer features five distinct I/O modes which when combined with an extensive set of I/O instructions provides a very powerful and easy to use I/O structure. These modes are:

1. Programmed I/O via registers
2. Programmed I/O via memory
3. Automatic I/O under interrupts
4. Block I/O
5. DMA

Transfers can be made to or from the A or X registers or directly to or from memory, whichever is more convenient. Both word and byte data can be handled directly, with byte data being packed automatically, if desired, without the need for time and space-consuming programmed routines.

1.3.1 Programmed Input/Output via Registers

For greater convenience in handling data that must be examined immediately upon input or is the result of computations that must be output immediately, programmed I/O transfers the data directly to and from the operating registers of the computer. Furthermore, programmed I/O instructions can be combined with sense and skip instructions to allow testing of controller or peripheral status prior to making a transfer.

1.3.2 Programmed Input/Output via Memory

This mode capitalizes on the power of the Automatic I/O instructions to transfer data to or from memory without disturbing the working registers of the computer. Any size block of data may be transferred into or out of memory at any address.

1.3.3 Automatic Input/Output under Interrupt Control

This mode permits an interface to transfer data to or from memory at its own data rate with minimal disturbance of the main program. When all data has been transferred, the interface develops an end-of-block interrupt in response to the Processor ECHO signal which, in turn, causes an interrupt subroutine to be entered which performs the necessary housekeeping associated with end-of-block operations.

1.3.4 Block Input/Output

For high speed transfer rates, Block I/O transfers blocks of any length at rates up to 131,579 words per second. Data is exchanged directly between memory and the peripheral interface with the index register providing the word count. During Block I/O instructions, the computer is totally dedicated to the execution of the instruction and cannot respond to interrupts until the entire block has been transferred.

1.3.5 Direct Memory Access (DMA)

For very high speed transfer rates, the DMA handles data directly with the memory at rates up to 625,000 words per second. Since this data transfer does not require the Central Processor, it can be performing other operations while interleaving with DMA on a cycle stealing basis. Multiple DMA controllers may use the DMA feature simultaneously (interleaved cycles) up to the full memory transfer rate. When more than one memory module is installed, memories may be two way interleaved to provide data transfer at twice the individual memory data rates.

1.4 DATA TRANSFER RATES

The maximum data transfer rates that can be achieved with the ALPHA LSI computer are listed in Table 1-1.

Table 1-1. ALPHA LSI Maximum Data Transfer Rates

MODE	DATA TRANSFER RATE
DMA (Interleaved memories)	1,250,000 words per sec
DMA (single memory)	625,000 words per sec
BLOCK I/O	131,579 words per sec
Programmed I/O via registers	34,247 wds/bytes per sec
Programmed I/O via memory	24,631 wds/bytes per sec
Automatic I/O under interrupts	26,738 wds/bytes per sec

Section 2
EXTERNAL I/O BUS CHARACTERISTICS

2.1 INTRODUCTION

This section describes the signals and the electrical characteristics of the NAKED MINI 16 LSI Computer external I/O bus. Additionally, the distribution of the external I/O bus and the ALPHA 16 LSI Computer motherboard are discussed.

2.2 EXTERNAL I/O BUS COMPONENTS (Figure 2-1)

The NAKED MINI 16 LSI Computer external I/O bus consists of three major components: the Address bus (A), the Data bus (D), and the Control bus (C).

2.2.1 Address Bus (A)

The Address bus consists of 16 lines (AB00- through AB15-) that are time shared by the Processor and DMA controllers.

The Processor and DMA controllers use the 15 bits of the A bus to address memory locations. The 16th A bus bit is used to specify word or byte memory operation. During I/O operations, the Processor uses the low order 8-bits of the A bus to convey device address and function code information to I/O devices. The high order 8-bits contain random information and are not normally used. The format of the low order 8-bits during I/O operations is as follows:

AB07-	Device Address bit 4	
AB06-	Device Address bit 3	
AB05-	Device Address bit 2	P Field
AB04-	Device Address bit 1	
AB03-	Device Address bit 0	
AB02-	Function Code bit 2	
AB01-	Function Code bit 1	F Field
AB00-	Function Code bit 0	

NOTE

The eight lines devoted to the Device Address and Function Code are arbitrarily divided into groups of five and three, respectively. They can be divided differently to increase or decrease the number of device addresses and function codes. For example, six lines can be devoted to the Device Address and only two to the Function Code. This would increase the number of device addresses to 64 and reduce the number of function codes to 4.

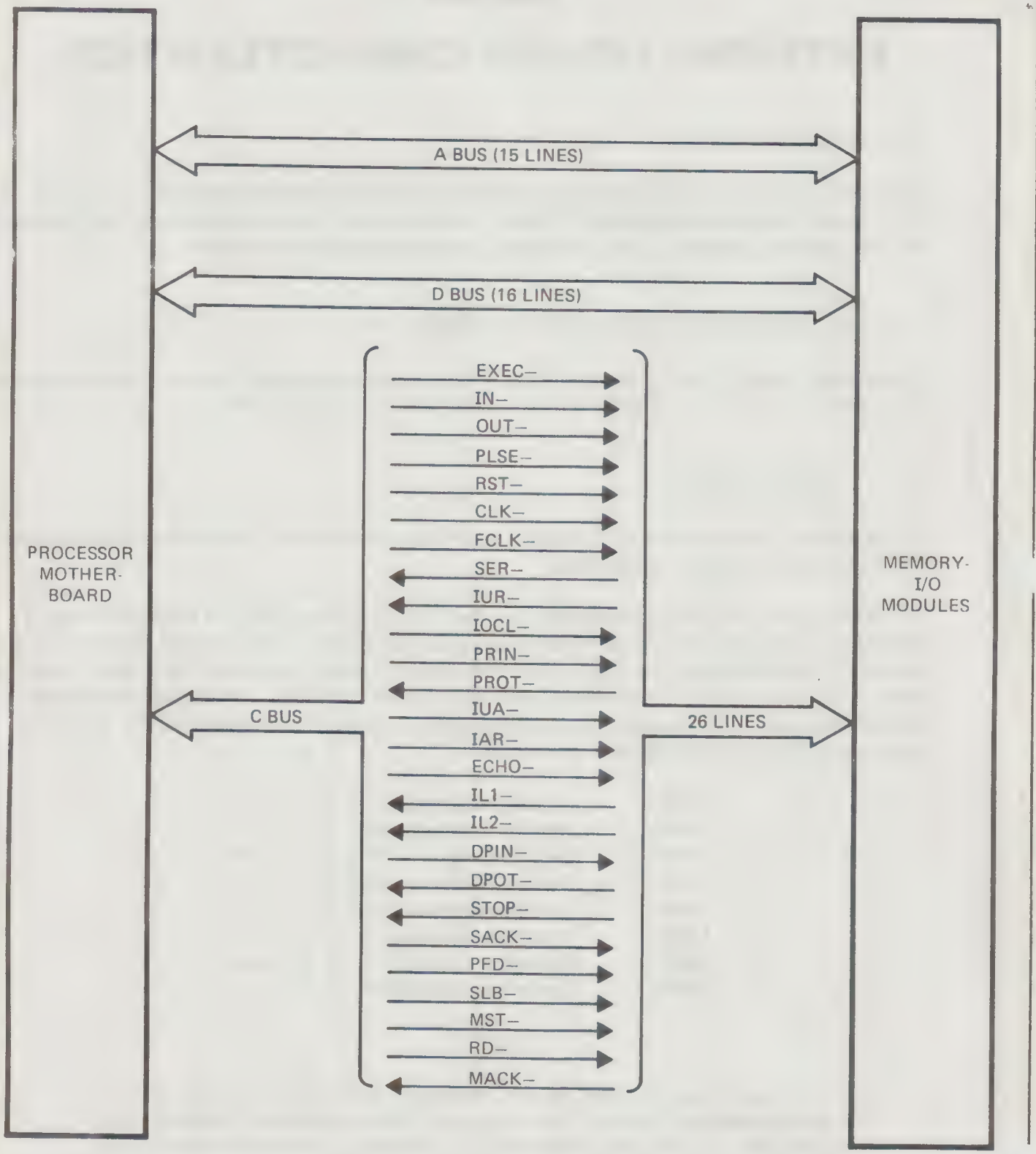


Figure 2-1. I/O Bus Components

Throughout the remainder of this design guide, all examples which involve I/O addresses assume the arbitrary five and three division.

2.2.2 Data Bus (D)

The Data bus consists of 16 bidirectional lines (DB00- through DB15-) that are time shared by the Processor, memory, and I/O interface controllers.

The Processor uses the D bus to read data from or write data into memory. Likewise, the Processor uses the D bus to transfer data to or from an I/O interface controller.

A DMA controller uses the D bus to read data from or write data into memory.

I/O interfaces use the D bus to convey an interrupt address to the Processor during interrupt processing.

2.2.3 Control Bus (C)

The C bus consists of 26 unidirectional control lines which define the specific action that an interface device is to perform. Eighteen lines are outputs from the Processor to all memories and interface controllers while eight lines are inputs from either memories or interface controllers to the Processor. The 26 C bus lines are subdivided into four broad categories: I/O command, utility signals, interrupt signals and DMA signals. Except as noted below, all Processor generated or received signals may also be generated or received by DMA controllers during DMA operations.

2.2.3.1 I/O Commands

There are three signals in this category: EXEC-, IN- and OUT-. These signals define the type of I/O operation in process.

EXEC- Execute is a Processor generated signal that indicates that the current instruction is a Select or Select and Present instruction. EXEC- is used typically to set or reset controls in the addressed interface.

IN- Input is a Processor generated signal that indicates that the current instruction is an input instruction and that the addressed interface should place input data on the data bus.

OUT- Output is a Processor generated signal that indicates that the current instruction is an output instruction and that the Processor has placed output data on the data bus for the addressed interface controller to accept.

2.2.3.2 Utility Signals

There are five signals in this category: PLSE-, RST-, CLK-, FCLK- and SER-.

PLSE- Pulse is a Processor generated signal which is used as a strobe pulse to load registers during an output transfer, set or reset controls during a Select instruction, reset data transfer controls during an input transfer, and to reset Interrupt Stimulus Store controls upon recognition of an interrupt.

RST- System Reset is a Processor or Console generated signal which is used to reset all controls in ALL interfaces to a known starting configuration. RST- is generated by the Processor in response to a power failure condition, an Autoload Initiation sequence, or when the Console RESET switch is depressed. Note - not driven by DMA controllers.

CLK- Clock is a Processor generated, 1 megahertz, free-running square wave signal that may be used as a timing reference by interface controllers--it is not synchronized to Processor operations. Note that only the Processor generates this signal. DMA controllers may not generate this signal.

FCLK- Fast clock is a Processor generated, 10 megahertz, free running square wave signal that is used as a timing reference by DMA controllers--it is not synchronized to Processor operations.

SER- Sense Response is a signal generated by an addressed interface controller to convey status to the Processor.

2.2.3.3 Interrupt Signals

There are nine signals associated with interrupt generation and processing. These signals are: IUR-, IOCL-, PRIN-, PROT-, IUA-, ECHO-, IL1-, and IL2-.

IUR- Interrupt Request is a multiplexed interrupt request line which multiple interfaces generate to request interrupt service. All interfaces which use this line are forced to compete with each other for recognition by the Processor. If two or more interfaces request interrupt service at the same time, recognition is given to the highest priority interface via the priority string (PRIN- and PROT-).

IOCL- I/O Clock is a Processor generated signal which is used by interfaces to synchronize IUR interrupt requests to the Processor. IOCL has a minimum duration of 150 nanoseconds; however, the duration varies with internal Processor operation. When an interrupt is recognized by the Processor, IOCL is inhibited to prevent the generation of additional IUR interrupt requests. IOCL remains inhibited until the Processor completes execution of the interrupt instruction.

PRIN- and PROT- Priority In and Priority Out. PRIN- and PROT- form an interrupt priority chain which is strung serially through all interface controllers and memories. PRIN- is the name given to the priority chain where it enters an interface. If low, it allows the interface to generate interrupts. Each interface generates a PROT- signal to indicate that neither it nor other upstream devices are generating an interrupt. The PROT- signal from each interface is the PRIN- signal for the next downstream interface.

IUA- Interrupt Acknowledge is a Processor generated signal which goes true upon recognition of any interrupt and remains true during execution of the interrupt instruction.

IAR- Interrupt Address Request is a Processor generated signal which is used to request an interrupt address from an interface in response to an interrupt request.

ECHO- Echo is generated by the Processor when an Auto I/O instruction has transferred all data or by an IMS instruction when the count overflows. ECHO- is typically used by the interface to request an interrupt. This interrupt vectors to a user-determined location in memory which normally contains a JST instruction to a subroutine which performs the necessary housekeeping associated with an end-of-block or elapsed count operation.

IL1- and IL2- Interrupt Lines 1 and 2 are interface generated high priority interrupt request lines which interrupt to locations : 0002 and : 0006, respectively. They are higher priority than the IUR line. IL1 has priority over IL2. IL1 and IL2 do not require interrupt vectoring by the interface as does IUR.

2.2.3.4 DMA Signals

Nine signals are associated with DMA control and processing. These signals are: DPIN-, DPOT-, STOP-, SACK-, PFD-, SLB-, MST-, RD- and MACK-.

DPIN- and DPOT- DMA Priority In and DMA Priority Out. DPIN and DPOT form a DMA priority chain which is strung serially through all DMA controllers and memories. DPIN- is the name given to the priority chain where it enters a DMA controller. If low, it allows the controller to access memory. Each controller generates a DPOT- signal to indicate that neither it nor other upstream controllers are communicating with memory. The DPOT- signal from each controller is the DPIN- signal for the next downstream controller. The DPIN- and DPOT- signals are strong through the "200" side of the motherboard only (see paragraph 2.4).

STOP- Stop Processor is a DMA controller generated signal which stops the Processor upon completion of its current machine cycle to permit the DMA controller to gain control of the I/O bus. STOP- may be generated at any time and may remain active for any length of time.

SACK- Stop Acknowledge is a negative-true Processor generated signal which informs DMA controllers that the Processor is no longer controlling the I/O bus and that the DMA controllers may take over the I/O bus. SACK- will remain true until STOP- is removed.

PFD- Power Failure Detected is a power supply generated signal which when active forces any DMA operations to terminate in order to permit the Processor to shut down the system in an orderly manner.

SLB- Select Least Significant Byte is a Processor or DMA controller generated signal which is used for Byte Mode memory accesses. When SLB- is low, the least significant byte (bits 0 through 7) of the addressed memory word is accessed. When SLB- is high, the most significant byte (bits 8 through 15) of the addressed memory word are accessed. SLB- is used to disable memory during Autoload operations by forcing it low while AB15- is high (word mode).

MST- Memory Start is a Processor or DMA controller generated signal which is used to initiate a memory cycle.

RD- Read Mode is a Processor or DMA controller generated signal which when low indicates the current memory cycle is a read/restore cycle. When high RD- indicates that the current memory cycle is a clear/write cycle. The Memory Protect option will force RD- low during any attempt to write into a protected memory location.

MACK- Memory Acknowledge is a memory generated signal that is used to inform the Processor or DMA controller that data is available during a read operation or that data has been accepted during a write operation.

Table 2-1. External I/O Bus Load, Drive and Termination Summary

			DEVICE TYPE(S) (Refer to Note 1)		
	SIGNAL	PIN	CPU	I/O CONT	DMA CONT
	GND	1			
	GND	2			
	+12	3			
	+12	4			
	+12	5			
	+12	6			
	-12	7			
	-12	8			
Note 2	DPIN-	9		J	5
Note 2	DPOT-	10	GND	J	4
Note 3		11			
Note 3		12			
	+5	13			
	+5	14			
	MST-	15	1,6		
		16			1
	MACK-	17	5,6		
	RD-	18	2,6		5
	FCLK-	19	3	5	2
	SLB-	20	2,6		5
	PFD-	21	5,6		1
	MDIS-	22	3		5
	AB08-	23	1		
	AB09-	24	1		1
	AB10-	25	1		1
	AB11-	26	1		1
	GND	27			1
	GND	28			
	AB12-	29	1		
	AB13-	30	1		1
	AB14-	31	1		1
	AB15-	32	1,6		1
					1
Note 3		33			
Note 3		34			
	STOP-	35	5,6		
	SAC-	36	3		2
Note 2	MBIN-	37		J	5
Note 2	MBOT-	38	4	J	J
	DB00-	39	1,5,6	2,5	J
	DB01-	40	1,5,6	2,5	1.5
	DB02-	41	1,5,6	2,5	1,5
	DB03-	42	1,5,6	2,5	1,5
	+5	43			1.5

2.3 ELECTRICAL CHARACTERISTICS

The external bus is best classified as a hybrid tri-state open-collector (wire-OR) bus, unterminated.

Most Processor drivers are tri-state power elements, capable of sinking 32 mA at 0.4 Vdc maximum and sourcing 2.0 mA at 2.4 Vdc minimum. In a few isolated cases, open-collector TTL drivers (32 mA sink at 0.4 Vdc) are used.

Processor receivers present one standard TTL load to the line (-1.6 mA at 0.8 Vdc, 40 μA at 2.4 Vdc). Depending on the nature of the particular signal, pullup resistors to +5 Vdc are used.

Open-collector drivers in I/O and memory modules are permitted on those bus lines for which pullup resistors are provided. Minimum required drive capablility is -32 mA at 0.4 Vdc max. Tri-state drivers electrically equivalent to the Processor bus drivers are also allowed, as long as the logic design of the system guarantees that no two tri-state drivers connected to the same bus line are simultaneously enabled. Receivers on I/O and memory modules may be any standard 74 series TTL device. Only one such receiver per module is permitted. Maximum loading shall not exceed 1.8 mA per module.

Logic Levels

(Negative-true)

logic "1": +0.4 Vdc max.
logic "0": +2.4 Vdc min

Table 2-1 summarizes the driver, receiver and pullup circuits.

2.4 MOTHERBOARD ORGANIZATION

Any slot (other than the slot dedicated to the NAKED MINI 16 LSI card) can accept either an I/O or memory module.

Figure 2-2 provides a diagram of the system motherboard. The motherboard provides for six slots used as follows:

Slot	Purpose
A	NAKED MINI 16 LSI Computer
B	Universal (Memory or I/O)
C	Universal (Memory or I/O)
D	Universal (Memory or I/O)
E	Universal (Memory or I/O)
F	Power Supply

Table 2-1. External I/O Bus Load, Drive and Termination Summary

			DEVICE TYPE(S) (Refer to Note 1)		
	SIGNAL	PIN	CPU	I/O CONT	DMA CONT
	GND	1			
	GND	2			
	+12	3			
	+12	4			
	+12	5			
	+12	6			
	-12	7			
	-12	8			
Note 2	DPIN-	9		J	5
Note 2	DPOT-	10	GND	J	4
Note 3		11			
Note 3		12			
	+5	13			
	+5	14			
	MST-	15	1,6		1
		16			
	MACK-	17	5,6		5
	RD-	18	2,6		2
	FCLK-	19	3	5	5
	SLB-	20	2,6		1
	PFD-	21	5,6		5
	MDIS-	22	3		
	AB08-	23	1		1
	AB09-	24	1		1
	AB10-	25	1		1
	AB11-	26	1		1
	GND	27			
	GND	28			
	AB12-	29	1		1
	AB13-	30	1		1
	AB14-	31	1		1
	AB15-	32	1,6		1
Note 3		33			
Note 3		34			
	STOP-	35	5,6		2
	SAC-	36	3		5
Note 2	MBIN-	37		J	J
Note 2	MBOT-	38	4	J	J
	DB00-	39	1,5,6	2,5	1.5
	DB01-	40	1,5,6	2,5	1,5
	DB02-	41	1,5,6	2,5	1,5
	DB03-	42	1,5,6	2,5	1.5
	+5	43			

Table 2-1. External I/O Bus Load, Drive and Termination Summary (Cont'd)

		DEVICE TYPE(S) (Refer to Note 1)		
SIGNAL	PIN	CPU	I/O CONT	DMA CONT
+5V	44			
DB04-	45	1,5,6	2,5	1,5
DB05-	46	1,5,6	2,5	1,5
DB06-	47	1,5,6	2,5	1,5
DB07-	48	1,5,6	2,5	1,5
DB08-	49	1,5,6	2,5	1,5
DB09-	50	1,5,6	2,5	1,5
DB10-	51	1,5,6	2,5	1,5
DB11-	52	1,5,6	2,5	1,5
DB12-	53	1,5,6	2,5	1,5
DB13-	54	1,5,6	2,5	1,5
DB14-	55	1,5,6	2,5	1,5
DB15-	56	1,5,6	2,5	1,5
EXEC-	57	1,6	5	5
IN-	58	1.6	5	5
GND	59			
GND	60			
IOCL-	61	1,6	5	5
OUT-	62	1,6	5	5
CLK-	63	3	5	5
SER-	64	5,6	2	2
IUR-	65	5,6	2	2
IL1-	66	5,6	2	2
IAR-	67	1,6	5	5
IL2-	68	5,6	2	2
RST-	69	2,5,6	5	5
IUA-	70	1,6	5	5
PLSE-	71	1,6	5	5
ECHO-	72	1,6	5	5
+5V	73			
+5V	74			
AB00-	75	1	5	1,5
AB01-	76	1	5	1,5
AB02-	77	1	5	1,5
AB03-	78	1	5	1,5
AB04-	79	1	5	1,5
AB05-	80	1	5	1,5
AB06-	81	1	5	1,5
AB07-	82	1	5	1,5

Table 2-1. External I/O Bus Load, Drive and Termination Summary (Cont'd)

		DEVICE TYPE(S) (Refer to Note 1)		
SIGNAL	PIN	CPU	I/O CONT	DMA CONT
PRIN-	83		5	5
PROT-	84	4	4	4
GND	85			
GND	86			

NOTES: 1. Device types are as follows-
1) Tri-state driver
2) 32 mA open-collector driver (7438 or equiv.)
3) 32 mA TTL driver (7437 or equiv.)
4) 16 mA TTL driver (7400 or equiv.)
5) TTL receiver (7404 or equiv.)
6) Pullup resistor (1K ohm)
J) Jumper

2. DPIN-, DPOT-, MBIN- and MBOT- are strung through the 200 series connectors only. These pin positions are unassigned on the 100 series connectors and are reserved for future expansion.

3. These pins are unassigned on both the 100 and 200 series connectors and are reserved for future expansion.

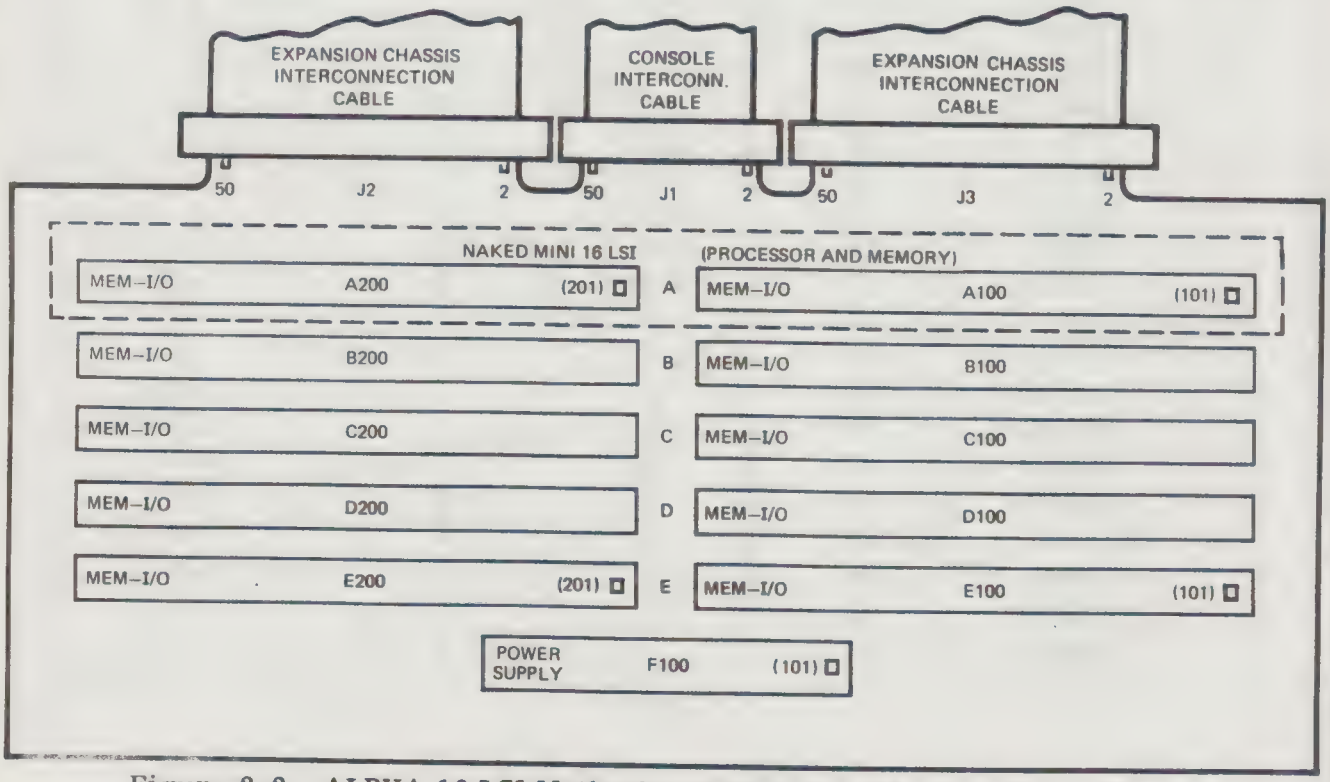


Figure 2-2. ALPHA 16 LSI Motherboard Slot Organization (Rear View)

In any given slot, either a full-width card (15" x 16.5") or two half-width cards (each 7.5" x 16.5") may be installed. One slot contains two connectors. The pins on one connector of a slot are numbered 100 through 186, and is referred to as the 100-series connector; similarly, the other connector (pins 200 through 286) is referred to as the 200-series connector.

With the exception of the priority chains, memory bank control, and two special CPU power supply signals, all signals are wired in a U fashion through all half-card connectors. All exceptions are described below.

2.4.1 Interrupt Priority

The daisy-chained interrupt priority string (PRIN-, PROT-) is wired in S fashion beginning at the 100-series connector of slot A, across to the 200-series connector, then in reverse direction across the two B slot connectors, etc., until all slots are connected. Both ends of the chain are connected to the expansion connectors. Both PRIN- and PROT- on the CPU connector (100-series connector, slot A) are used to carry special signals to the console; the actual origin of the priority chain is therefore the memory side of the top slot (slot A100).

2.4.2 Memory Bank Control, DMA Priority

The memory bank control (MBIN, MBOT) and DMA priority (DPIN-, DPOT-) daisy-chains run down the 200-series connectors only. Therefore, memories and DMA controllers must be either full cards or half cards installed on the 200 series side only.

2.4.3 CPU Power Supply Signals

Two lines from the power supply, TTLF (twice the line frequency) and +5 Hz are wired directly between the power supply slot and CPU slot (slot A100).

2.5 EXPANSION

To facilitate expansion of the computer system beyond the first chassis and to provide for interconnect to the ALPHA 16 LSI console, expansion connectors are supplied on the motherboard immediately above slot A. Two connectors are provided for I/O bus expansion and one connector is provided to interconnect the console.

In the event that insufficient slots are provided in the basic ALPHA 16 LSI chassis for a given application, the external bus may be expanded via a buffer card, cables, and expansion chassis.

The expansion chassis is a second ALPHA 16 LSI chassis (identical motherboard, etc.). A buffer card is required to regenerate the external bus. Cables interconnect the buffer card to the two motherboards via the expansion connectors. The external bus of the computer on the motherboard may not be extended without electrical buffering.

Expansion may extend to two, three or more chassis. As expansion chassis are installed, a speed degradation will occur. Memory modules located in expansion chassis will exhibit an apparent slower system access time. Similarly, I/O modules located in a second expansion chassis or beyond may require that the timing circuit of the computer be altered to provide additional phase stretching during I/O operations. This timing circuit is modified simply by changing an option-jumper connector which configures all jumper-controlled Processor options in the machine. This option-jumper connector mounts to the read-edge of the computer option card.

2.6 NAKED MINI 16 LSI EXTERNAL BUS REQUIREMENTS

In applications where the NAKED MINI 16 LSI computer is used without the system motherboard and is instead connected to I/O and/or memory modules via user-supplied cabling, printed circuit card, etc., the line length of each signal must be limited to 18 inches.

The external bus must be designed to minimize crosstalk, reflections, etc., so as to preserve signal integrity. Recommendations as to line termination are available upon request. In general, consultation with Computer Automation is recommended to ensure system performance.

2.7 TWO-MODULE OPTIONS

Any option requiring more than one printed circuit card may not use the motherboard for interconnection. Unique interconnections may be made via a jumper cable installed on the rear edge of the two cards.

Section 3
I/O TRANSFER TIMING

3.1 INTRODUCTION

This section describes the I/O transfer timing of the various I/O instructions. I/O transfer timing is the period during an I/O instruction when data is transferred between the Processor and an interface and is shown in Figure 3-1.

NOTE

Unless otherwise noted, all timing intervals indicated in timing diagrams are given in nanoseconds. All timing intervals discussed in text are nominal.

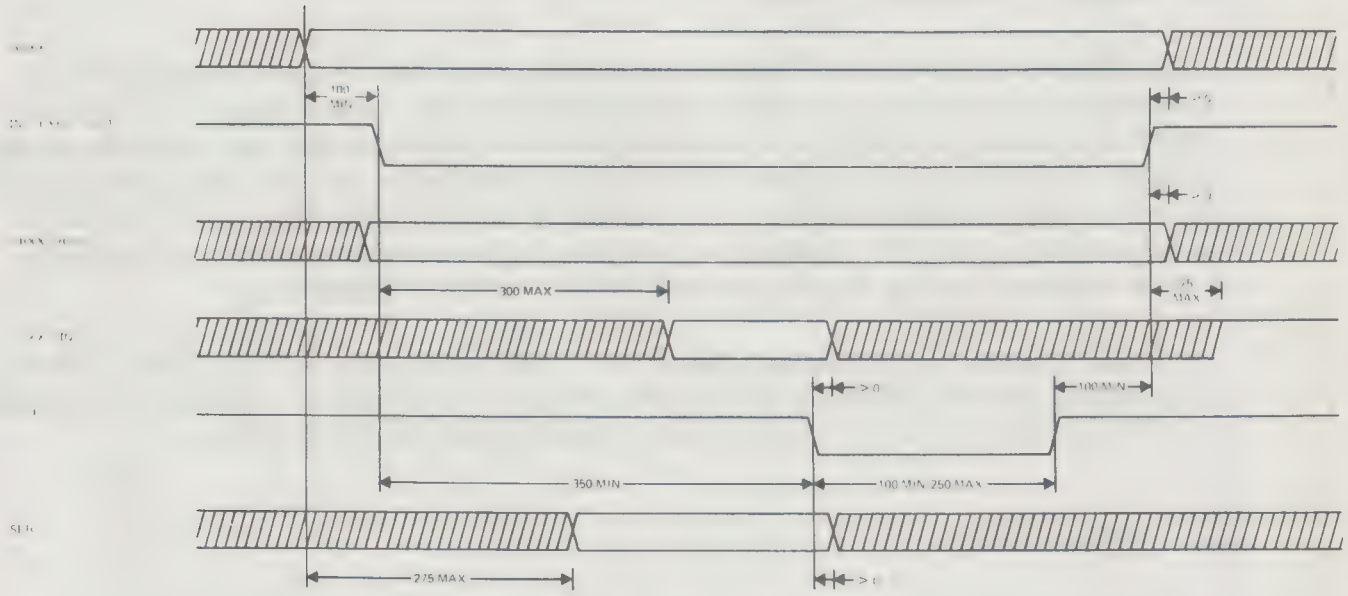


Figure 3-1. I/O Transfer Timing

3.2 I/O BUS CONSIDERATIONS

The A bus is active for non-I/O as well as I/O instructions. To guard against responding to a non-I/O instruction, the I/O control signals (EXEC-, IN- or OUT-) should be used when interpreting the A bus. The SER- signal is the only exception and may be driven independent of EXEC-, IN- or OUT-.

Data should never be placed on the D bus by an interface except in the presence of IN- or IAR-.

3.3 SENSE COMMAND TIMING

No I/O bus control signals are generated by the Processor during a Sense instruction. The addressed interface controller uses the function code information to determine which one of eight possible functions are to be sensed. The sense information is sent to the Processor via the SER- line. If the Processor is looking for a sense response, the SER- signal is gated into the Processor, otherwise it is ignored. The user has 275 nanoseconds to stabilize the sense response after receipt of the Device Address signals.

3.4 SELECT COMMAND TIMING

During Select or Select and Present commands, the EXEC- signal is generated a minimum of 100 nanoseconds after the A bus stabilizes. The D bus is selected for output as a result of EXEC- and becomes stable at the same time as the leading edge of EXEC-. If a command register is used, the information on the D bus can be presented to the register by EXEC- and clocked in with PLSE-. The D bus contains all zeros during the SEL instruction and is equal to the contents of the Processor A or X register during the SEA or SEX instructions, respectively.

The PLSE- signal is developed a minimum of 350 nanoseconds after EXEC-. PLSE- is generally used to clock all control flip-flops in the interface. Either the leading or trailing edge of PLSE- may be used to set or reset control flip-flops.

3.5 INPUT TIMING

All input sequences, regardless of the input command type, appear basically the same to an interface. For all Input commands, the IN- signal is generated a minimum of 100 nanoseconds after the A bus stabilizes. The D bus is selected for input as a result of IN-. The IN- signal is used by the interface to gate data onto the D bus. Data must be present and stable on the D bus no later than 300 nanoseconds after IN- goes low.

The PLSE- signal is developed a minimum of 350 nanoseconds after IN- goes low. PLSE- is typically used to reset the buffer ready control in the interface. Either the leading or trailing edge of PLSE- may be used to reset the buffer ready control. Note, however, that data on the D bus must remain stable until the leading edge of PLSE- and must be removed no later than 75 nanoseconds after the trailing edge of IN-.

If the Input command issued is conditional, the sense response (SER-) must be stable no later than 275 nanoseconds after the A bus stabilizes to guarantee detection of SER- by the Processor. If SER- is high from the 275 nanosecond point to the leading edge of PLSE-, the entire input sequence is repeated for a conditional input or block input until the SER- line goes low. If SER- is low at the 275 nanosecond point, the operation is terminated after the present cycle and PLSE- is generated to indicate that the Processor has accepted the data. If SER- changes state between the 275 nanosecond point and the leading edge of PLSE-, the Processor may or may not detect SER-.

All sense responses are ignored by the Processor when executing unconditional Input commands.

3.6 OUTPUT TIMING

All output instruction sequences, regardless of the Output command type, appear basically the same to an interface. During an Output command, the OUT- signal is generated a minimum of 100 nanoseconds after the A bus stabilizes. The D bus is selected for output as a result of OUT-. Once selected, the D bus stabilizes at the same time as the leading edge of OUT-.

The PLSE- signal is generated a minimum of 350 nanoseconds after OUT- goes low. PLSE- serves two functions. The first is to clock output data into a receiving register in the interface. The second function is to reset the output buffer empty control in the interface.

If the Output command is conditional, the sense response must be stable no later than 275 nanoseconds after the A bus stabilizes to guarantee detection of SER- by the Processor. If SER- is high from the 275 nanosecond point to the leading edge of PLSE-, the entire output sequence is repeated until the SER- line goes low. If SER- is low at the 275 nanosecond point, the operation is terminated after the present cycle and PLSE- is generated to indicate the availability of data to the interface. If SER- changes state between the 275 nanosecond point and the leading edge of PLSE-, the Processor may or may not detect SER-.

Any sense responses that are generated during an unconditional Output command are ignored by the Processor.

3.7 AUTOMATIC INPUT AND OUTPUT TIMING

The Automatic Input and Output commands have essentially the same transfer timing as all other I/O commands. The only difference is that when used as interrupt instructions they develop an ECHO- signal to the interface when the last word or byte of data has been transferred. The ECHO- signal occurs a minimum of 350 nanoseconds after IN- or OUT- during the last transfer. ECHO- is typically used by the interface to develop an end-of-block interrupt. These commands are unconditional commands that do not require a sense response.

Section 4
INTERRUPT CHARACTERISTICS

4.1 INTRODUCTION

Minicomputers perform in a wide variety of applications where they communicate with many different types of devices. These devices operate at widely varying speeds and generate events that occur randomly rather than at evenly spaced time intervals. If the events do occur at evenly spaced time intervals, these intervals may be relatively far apart. For these reasons, a versatile and efficient computer needs a priority interrupt system.

If a computer does not have a priority interrupt system, the computer must poll all of the external devices which may require service. The polling must be at frequent enough intervals so that events are serviced within a reasonable time after they occur. Polling consumes considerable time, and may not allow much processing time between the handling of external events.

A priority interrupt system relieves the computer of the polling responsibility. The computer may continue processing data between external events, and may take time out from main program processing to handle external events as they occur.

The ALPHA LSI computer features five levels of interrupts. Each interrupt level uses an interrupt-request line to get the attention of the Processor. Upon getting the attention of the Processor, the interrupt source vectors the Processor to an interrupt location in memory. The interrupt location contains an interrupt instruction which defines the specific action that the Processor is to take in processing the interrupt.

The interrupt request lines are designated: Power Fail Interrupt (PFI); Console (TRAP) Interrupt (CINT); Interrupt Line 1 (IL1); Interrupt Line 2(IL2); and Interrupt Request (IUR). A priority level exists between each of these lines wherein PFI has the highest priority, CINT is second, IL1 is third, IL2 is fourth and IUR is lowest in priority. PFI, CINT, IL1 and IL2 are self-vectoring lines (the user does not have to supply the interrupt address). The IUR line is shared by multiple devices and features a priority chain to resolve priority when two or more devices issue an IUR interrupt request at the same time. Each of the PFI, CINT, IL1, IL2 and the multiple interrupt sources that share the IUR line cause the Processor to be vectored to distinct locations that can be anywhere in memory.

4.2 INTERRUPT LINES

The characteristics of each of the five interrupt request lines are discussed in the following paragraphs.

4.2.1 Power Fail Interrupt

The Power Fail Interrupt (PFI) services the power down interrupt only. PFI is the highest priority interrupt line in the interrupt system and is not accessible to the user via the I/O bus structure of the computer.

4.2.2 Console (TRAP) Interrupt

The Console (TRAP) Interrupt (CINT) services the console interrupt and trap interrupt only. CINT is the second highest priority interrupt line and is not accessible to the user via the I/O structure of the computer.

4.2.3 Interrupt Line 1

Interrupt Line 1 (IL1) vectors all interrupts to memory location : 0002. The memory parity option uses IL1, however, the parity error interrupt forces IL1 to vector to memory location : 0012. IL1 does not provide external priority resolution when servicing multiple devices. IL1 is the third highest priority interrupt line and is accessible to the user via the Processor I/O structure.

4.2.4 Interrupt Line 2

Interrupt Line 2 (IL2) vectors all interrupts to memory location : 0006. IL2 is the fourth highest priority interrupt line and is accessible to the user via the Processor I/O structure. Like IL1, IL2 does not provide external priority resolution to service multiple devices.

4.2.5 Interrupt Request

The Interrupt Request (IUR) vectors interrupts to the Processor from virtually an unlimited number of devices. The IUR line has a priority string associated with it. The priority string insures that a device with a higher priority will be serviced before a lower priority device when two or more IUR requests occur at the same time. When the interrupting device has priority, it must furnish an interrupt address to the Processor upon request. In general, IUR interrupt addresses are user defined.

4.3 PROCESSOR GENERATED INTERRUPTS

The ALPHA LSI computer generates two standard and six optional interrupts. In addition, two optional pseudo interrupts are generated. Each of these interrupts is discussed briefly in the following paragraphs in order of priority.

4.3.1 Power Fail/Restart Interrupt (Optional)

The Power Fail/Restart Option (PF/R) generates a power-down interrupt to location :001C whenever a low power condition exists. The power-down interrupt has the highest priority of any interrupt processed by the Processor. When power is restored to an acceptable level, the PF/R logic causes the program counter to be set to location :0000 and the RUN mode is established to restart the system. Although location :0000 is the power-up location, it is not a true interrupt location but rather a pseudo interrupt since no interrupt processing is required to get to location :0000.

4.3.2 Autoload (Optional)

The Autoload option utilizes the PF/R logic to develop a pseudo-interrupt to location :0000 of a special Autoload read-only-memory as a starting point for the autoload sequence.

4.3.3 Console Interrupt and Trap (Standard)

A console interrupt can be developed when the Processor is in the RUN mode and the AUTO switch on the console is depressed. A trap interrupt is developed when the TRP instruction is executed. Both the console and trap interrupt share the second-highest interrupt priority and they both interrupt to location :001E.

4.3.4 Memory Parity (Optional)

The Memory Parity option generates a parity error interrupt to location :0012 via the IL1 line whenever a parity error is detected during a memory read operation. The parity error interrupt is the highest priority interrupt on the IL1 line but is lower in priority than the power-down, console and trap interrupts.

4.3.5 Real Time Clock (Optional)

The Real-Time Clock (RTC) option generates a clock and sync interrupt. The clock and sync interrupts share the second highest priority on the IUR line. The clock interrupt is vectored to location :0018 while the sync interrupt is vectored to location :001A.

4.3.6 Teletype Interface (Optional)

The processor mounted TTY Interface generates both word and end-of-block interrupts via the IUR line. The word interrupt is vectored to location : 0002 while the end-of-block interrupt is vectored to location : 0006. These interrupt vectors are the same interrupt vectors that are used by the IL1 and IL2 lines. Since IL1 and IL2 do not provide priority resolution and are of a higher priority than these interrupts, the TTY word and end-of-block interrupts should be displaced to alternate locations when IL1 and IL2 are used. A jumper option permits the word and end-of-block interrupts to be displaced to locations : 0022 and : 0026, respectively. When used in the full duplex mode the teletype interface generates four interrupts (locations : 0002, : 0006, : 0022, and : 0026).

4.4 OFFSETTING PROCESSOR GENERATED INTERRUPTS

Figures 4-1 lists in the order of their absolute priority, the standard interrupt locations for all Processor generated interrupts. These interrupt locations are all located in the scratchpad area of memory. A jumper option permits the user to offset these locations by : 100 locations to place them outside the scratchpad area to allow more efficient utilization of the scratch area. IUR interrupts generated by non-processor mounted options may be individually offset to place them outside of the scratch area.

NOTE

The power-up restart and autoload start-up location (location : 0000) is not affected by the offset jumper option.

4.5 PERIPHERAL GENERATED INTERRUPTS

Peripheral interface controllers may request interrupt service via the IL1-, IL2- or IUR- request lines. The techniques used to develop these interrupt requests are discussed in detail in Section 5 of this design guide.

4.6 INTERRUPT TRANSFER TIMING (Figure 4-2)

For the purpose of priority resolution, all interrupts must be synchronized prior to being generated. Synchronization can occur only during a main-line program instruction. This is to insure that when executing the interrupt instruction, no other interrupt can intervene. When synchronization is obtained, the PROT- signal from the interrupting device goes high (false) to disable all down-stream IUR interrupts. When interrupts of higher priority than IUR are serviced, the Processor removes the PROT- signal (to the false state) to disable all IUR interrupts.

ABSOLUTE PRIORITY		INTERRUPT ADDRESS
1	POWER FAIL (PFI)	: 001C (: 011C)
2	TRAP INTERRUPT (CINT)	: 001E (: 011E)
3	CONSOLE INTERRUPT (CINT)	: 001E (: 011E)
4	MEMORY PARITY (IL1)	: 0012
5	INTERRUPT LINE 1 (IL1)	: 0002
6	INTERRUPT LINE 2 (IL2)	: 0006
7	RTC SYNC INTERRUPT (IUR)	: 001A (: 011A)
8	CLOCK INTERRUPT (IUR)	: 0018 (: 0118)
9	TTY END-OF-BLOCK (IUR)	: 0006 (: 0106); OPTIONAL : 0026 (: 0126)
10	TTY WORD (IUR)	: 0002 (: 0102); OPTIONAL : 0022 (: 0122)
11	SLOT B200	Slots B200 through E200 accommodate plug-in modules (either memory or I/O). All I/O modules may use the IUR line and must provide an interrupt address. Modules with multiple interrupt capabilities must have internal priority resolution and multiple addresses. The continuity of the priority chain must not be broken. If broken, interrupts below the break may not be recognized or may be recognized erroneously.
12	SLOT B100	
13	SLOT C100	
14	SLOT C200	
IUR CHAIN		
15	SLOT D200	
16	SLOT D100	
17	SLOT E100	
18	SLOT E200	
19	EXPANSION CHASSIS SLOT A100	
20	EXPANSION CHASSIS SLOT A200	
21	EXPANSION CHASSIS SLOT B200	

Figure 4-1. ALPHA LSI Interrupt Organization

If interrupts are enabled, the Processor recognizes an interrupt request when the current main-line program instruction has finished execution. When recognition of an interrupt is given, the Interrupt Acknowledge signal (IUA) is issued by the Processor and IOCL is turned off to inhibit any change in interrupt request status until the current interrupt operation is completed.

Approximately 2 microseconds after IUA- goes low, the Processor generates the Interrupt Address Request signal (IAR-) and selects the D bus for input. IAR- is used by the interface to generate the interrupt address. The IAR- signal is low for approximately 750 nanoseconds. During this interval, the user generated interrupt address must be available within 300 nanoseconds of IAR- and remain stable until the leading edge of PSLE-. PLSE- is used in the more complex interrupt structures to reset the Interrupt Stimulus Store Control.

IUA- will remain low until the interrupt instruction finishes execution. The duration IUA- is low is a function of the number of machine cycles that are required to execute the interrupt instruction. When IUA- goes high, IOCL is re-enabled permitting subsequent interrupts to be generated.

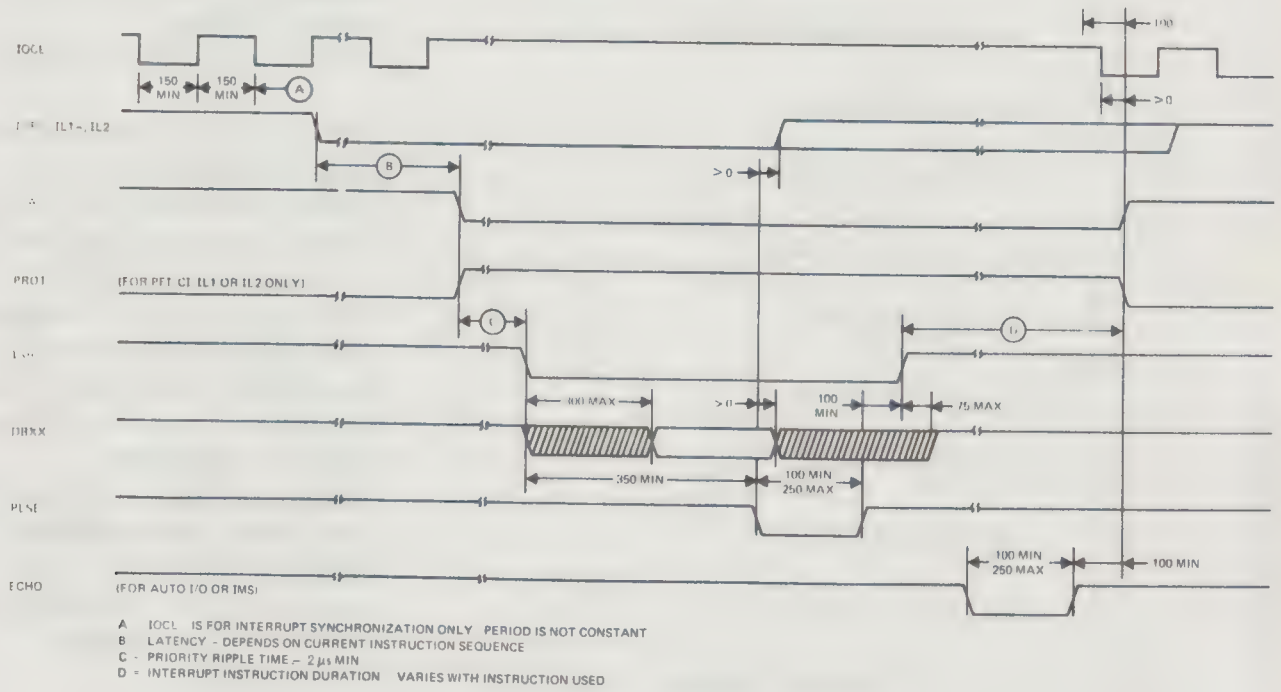


Figure 4-2. Interrupt Transfer Timing

4.7 INTERRUPT OPERATION CONTROL

Two levels of control are associated with IL1, IL2 and IUR interrupt processing--primary and secondary. The primary control level is provided by the Enable Interrupt flip-flop (EIN) in the Processor. The EIN flip-flop is accessible to the programmer and can be enabled or disabled on command. When enabled, EIN allows recognition of any interrupt. Likewise, when EIN is disabled, interrupts will not be recognized.

The secondary control level is provided by an interrupt enable flip-flop in each interface controller. The interrupt enable flip-flop enables or disables the interrupt structure of the interface controller. Like the EIN flip-flop discussed above, the interrupt enable flip-flop in each controller can be enabled or disabled by means of a select command addressed to the specific interface with the appropriate function code.

This dual system of interrupt control can be very useful to a programmer. With this system, the programmer can control interrupts in general with the EIN flip-flop, yet enable or disable interrupts from selected devices as conditions dictate.

Interrupts developed via the PFI and CINT lines are somewhat different in that they can be generated outside EIN control. In normal operation (that is, when operating under EIN control), the power fail, console and trap interrupts require that EIN be enabled. Most interrupt subroutines disable interrupts during execution of the subroutines causing high priority interrupts such as power fail to wait until EIN is re-enabled. A special jumper on the option board permits all interrupts generated on the PFI and CINT lines to be recognized regardless of the state of EIN.

When the jumper option is employed, two new instructions (PFE and PFD) are used to control the Power Fail circuits. The PFE instruction must have been issued before a Power fail interrupt can be generated. Likewise, the PFD instruction disables the generation of a power fail interrupt.

The Console Interrupt is controlled by the CIE and CID instructions in the same way as in normal operation. The trap interrupt is generated in the same manner as in normal operation. The only difference between normal operation and the jumper option is that EIN does not have to be set to generate the console and trap interrupts.

Another useful programming feature is the SIN instruction. The SIN instruction permits the programmer to suppress recognition of all interrupts (and Byte mode operation) for up to six instructions.

Once an interrupt structure is enabled, an interrupt can be generated in five basic steps:

Step 1 Stimulus Generation--The user generates the interrupt stimulus in response to some event or condition.

Step 2 Interrupt Request Generation--The interrupt structure of the interface controller, if enabled, stores the interrupt stimulus and generates an interrupt request.

Step 3 Interrupt Recognition--The Processor upon receipt of the interrupt request waits for the current instruction to finish execution, and if system interrupts are enabled (EIN set), issues an interrupt address request.

Step 4 Interrupt Address Generation--The interrupt structure of the interface controller responds to the interrupt address request by placing the interrupt address on the I/O data bus lines (except for IL1 and IL2 interrupt).

Step 5 Interrupt Instruction Execution--The Processor fetches and executes the instruction from the interrupt location.

4.8 INTERRUPT REQUEST LINE TRADE-OFFS

The user has a choice of three interrupt request lines, IL1, IL2 and IUR. The trade-offs associated with each of these lines are discussed below.

The IL1 and IL2 interrupt structures are the simplest structures to implement in terms of hardware since they do not require interrupt address logic, Processor synchronization logic, or down-stream priority disable logic. All of these functions are provided in the Processor. The IL1 and IL2 lines are intended for single device applications where high speed devices require the highest available priority

The IUR line is for multiple devices where each device competes for service via the priority chain. The priority of an interface controller can be changed by simply removing the interface controller from the computer chassis and relocating it in a higher or lower priority card slot. An IUR generating interface has greater flexibility in terms of address vectoring. If an address vector must be changed, the address may be offset from its base location to another location by means of address select lines.

Section 5

DEVICE INTERFACE CONTROLLER, DESIGN TECHNIQUES

5.1 INTRODUCTION

This section describes how to design a device interface controller that will be compatible with the I/O structure of the ALPHA LSI computer. The logic circuits described here are from Computer Automation, Inc. standard interface products that are successfully performing at user installations throughout the world.

5.2 I/O CONTROL IMPLEMENTATION

The following paragraphs describe device interface controller design requirements for compatibility with the I/O structure of the Processor.

5.2.1 Device Address Decoder (Figure 5-1)

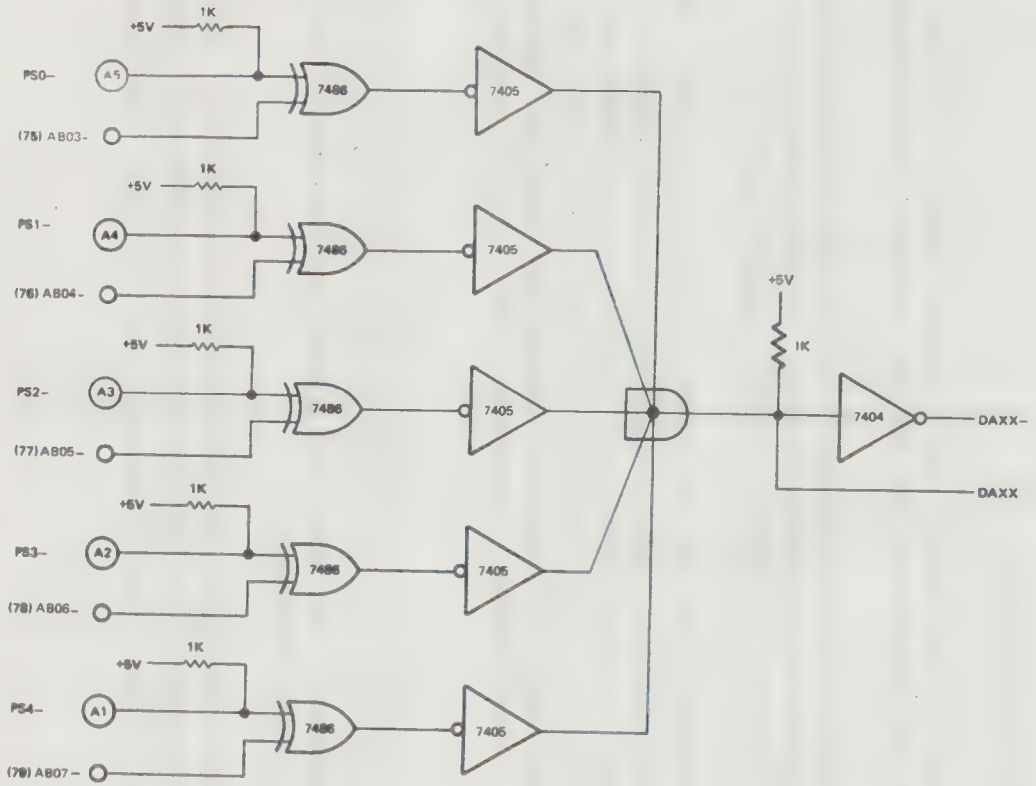
The device address decoder is a comparator circuit which compares the five-bit device address field of an I/O instruction with the user assigned device address.

The example A address decoder uses an exclusive-OR (EX OR) gate and an inverter for each of the five device address bits to be decoded. The outputs of the inverters are tied together to form a wired-AND address decoder output signal, DAXX.

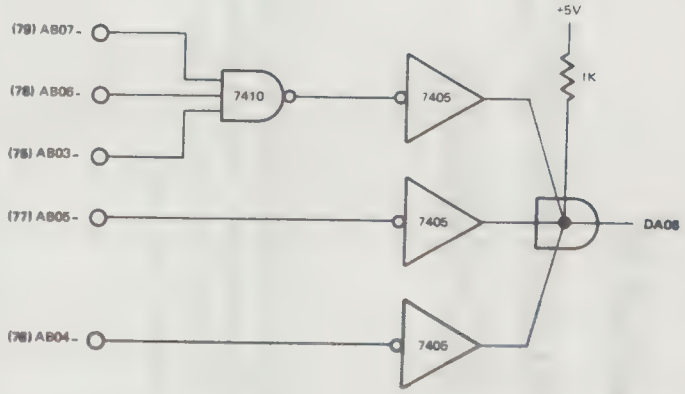
Address decoding is controlled by the five Peripheral Select signals (PS0- through PS4-). These signals are brought in from the peripheral interface connector to corresponding EX OR gates. If a true (low) address bit is to be decoded, the corresponding address select signal must be externally wired to ground (ground = true). Likewise, if a false address bit is to be decoded, the address select signal must be left open permitting the pull-up resistor to provide the false (high) address select signal.

When the device address bit agrees with the address select signal, the output of the EX OR gate is low. All five device address bits must agree with the user defined address selection. If agreement is obtained, the decoder output signal DAXX goes high enabling recognition of I/O commands.

Example B shows an address decoder which decodes device address : 6. This type of decoder is used only in dedicated applications and does not provide the flexibility that the example A decoder offers. Refer to Appendix A for standard device address assignments.



Example A. Non-Dedicated Application



Example B. Dedicated Application

Figure 5-1. Device Address Decoding Techniques

5.2.2 Function Decoder (Figure 5-2)

The Function Decoder uses an MSI chip, or a network comprised of SSI chips to decode the contents of the Function Bus. The result is a function code (1 of 8 maximum) which performs some function in the selected device interface controller.

The choice of chips depends upon the user's application. Figure 5-2 shows three examples, A, B and C of how to implement the function decoder. When decoding three or less functions, example C may be the most efficient. However, if chip count is a factor, example A or B is probably more efficient. In any case, where more than three functions are to be decoded, example A or B is probably the most efficient.

5.2.2.1 Example A

Example A uses a TTL 7442 MSI chip which is a 4 to 10 Decoder. Inputs A, B and C are the 2^1, 2^2, and 2^3 inputs respectively. Input D is the 2^4 input. When high, input D enables decoded output 8 or 9. However, only the first eight outputs of the decoder (0 through 7) are normally used, since eight is the maximum capacity of the three Function Bus lines in its normal configuration. D input is the enable input for the first eight decoded outputs, and utilizes the DAXX- signal for this purpose. When the device address is decoded, the DAXX- signal goes low, thus enabling the Function Decoder.

Input lines from the Function Bus are first unloaded by inverter gates and then applied to the decoder. As an example, if all Function Bus lines were false (high, implying Function Code 0), lows would be applied to inputs A, B and C. The decode of all low inputs would be zero thus causing FC0- to go low. (Decoded outputs of a 7442 are always low.) If a high signal is required, it can be obtained by using a simple inverter gate, such as the TTL 7404 illustrated.

5.2.2.2 Example B

Example B is the same as example A, except that the outputs are reversed (output 7 = FC0, output 6 = FC1, etc.). However, example B can only be used where the Function Bus lines will not be applied to any other circuit on the same device interface controller. This complies with the rule that each controller represents no more than one load to each I/O line.

5.2.2.3 Example C

Example C can decode only three function codes. TTL 7410 3-input NAND gates are the decoders. The three Function Bus signals are applied to the appropriate NAND gates to produce FC0- through FC2-. If the decoded device address is to enable the function codes, TTL 7420 NAND gates can be used, with the DAXX signal applied to the fourth input of each gate.

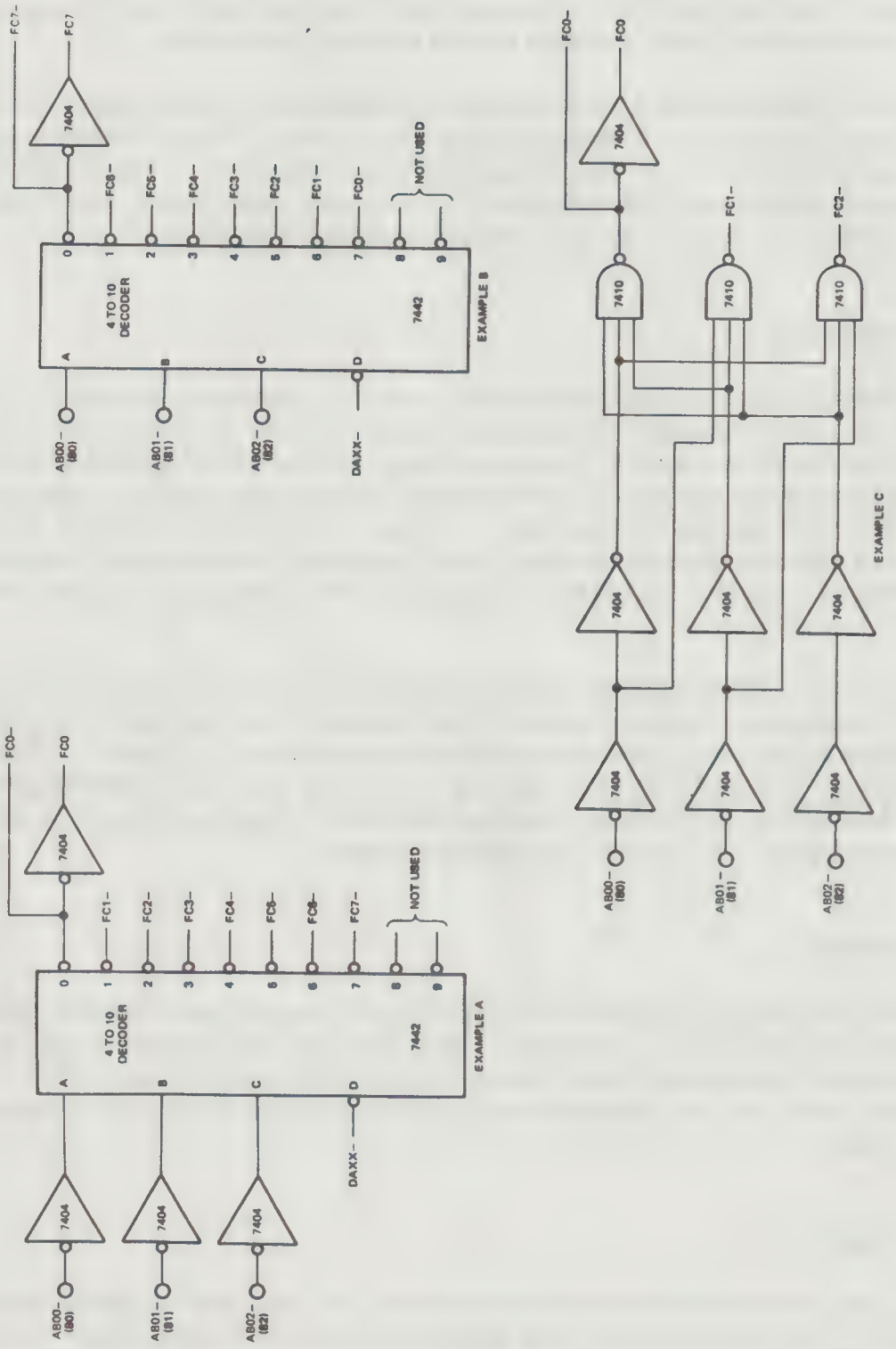


Figure 5-2. Function Decoder Configurations (Typical)

5.2.3 Select, Input or Output Command Decoding (Figure 5-3)

Similar to the Function Decoder, the Select, Input or Output (I/O) commands can be decoded by an MSI chip or a network of SSI chips. Figure 5-3 shows two methods, example A and B, of implementing this circuit. When the various commands are fully decoded using the F bus signals, the function bus decoder alone is not generally needed.

5.2.3.1 Example A

Example A shows a TTL 7442 4 to 10 Decoder used as a Select, Input or Output command decoder. The decoder also decodes the contents of the Function Bus, but only for the specific type of I/O command with which it is being used. Assume the decoder is used as a Select command decoder. The contents of the Function Bus are applied to the A, B and C inputs to produce the appropriate function code--any one of up to eight associated with the Select command. The decoder is enabled by NANDing DAXX (device address decoded), EXEC and PLSE. The Select command and associated functions are decoded by the one circuit. Refer to paragraph 2.4 for Select command timing.

5.2.3.2 Example B

Example B shows a decode network of SSI chips. This circuit can offer greater efficiency than the 7442 chip, depending upon the application. For example, if three types of commands (Select, Input and Output) are used by a controller, and less than three functions are associated with each type command, it is probably more efficient to use decoders of this type, each utilizing the outputs of a single Function Decoder.

5.2.4 Initialization Implementation (Figure 5-4)

Initialization circuitry establishes a known static state within a device interface controller. Initialization is started by executing a Select command with a function code dedicated to initialization (nominally Function Code 4) or by depressing the RESET switch on the Processor Control Panel. Figure 5-4 shows a circuit configuration for implementing initialization. When the device address and function code of the Select command are decoded, the DAXX and FC4 signals go high to prime the 3-input NAND gate. EXEC goes high during the Select command, enabling the gate to produce the INZX- and INZX signals. These signals are distributed throughout the controller to reset or set flip-flops, data registers, counters, etc., to establish the known static state. INZX is also produced when the RST- signal goes low upon depression of the RESET switch on the front panel, or during a power fail/restart situation.

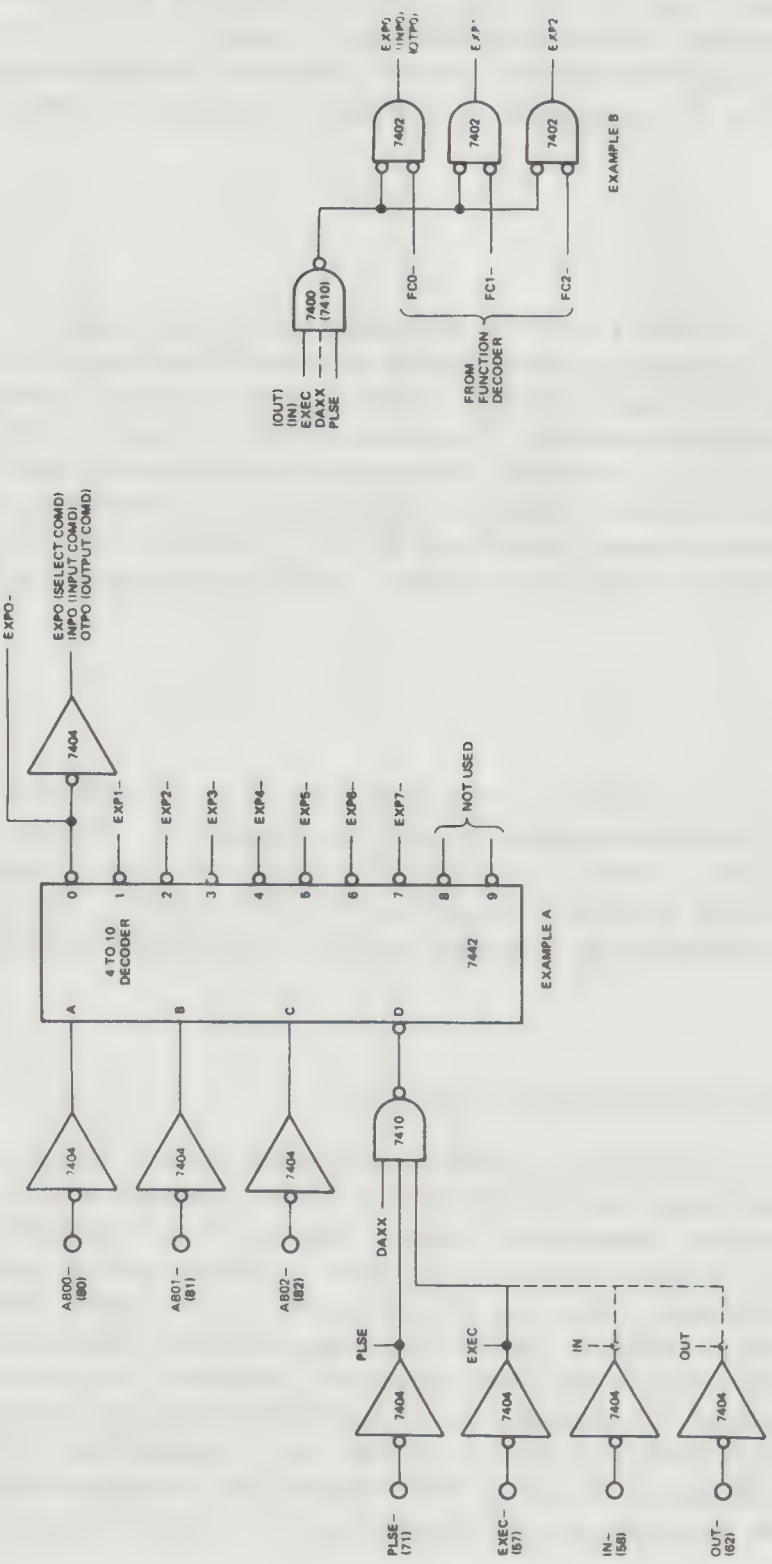


Figure 5-3. Select, Input or Output Command Decode Configurations

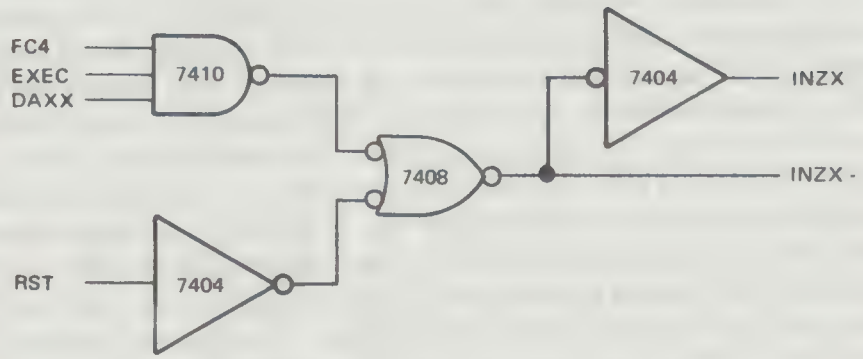


Figure 5-4. Initialization Circuit

5.2.5 Sense Command Implementation (Figure 5-5)

The Sense command circuit can be implemented using an MSI chip or a network comprised of SSI chips. As in the Function and I/O command decoders, application determines the most efficient method. An MSI chip can accommodate up to eight sense conditions, and provide its own function decoding. (Function code determines sense condition to be interrogated.)

The SSI network can be implemented more efficiently where three or less sense conditions are to be interrogated. However, the circuit requires inputs from a Function Decoder. Both positive and negative, internal and external signals can be sensed. An example of each is described below and illustrated in Figure 5-5.

5.2.5.1 Positive Sensing

Example A shows positive sensing using a TTL 74151 MSI chip. The 74151 is an 8 to 1 Multiplexer that provides internal function code decoding and an enable input (STROBE). It also provides both true and complement outputs. The top four inputs (0 through 3) are shown accepting External Sense (ES0 through ES3) signals from the external device. Pull-up resistors should be connected to each external input line (10K typical). Internal Sense (IS4 through IS7) signals are applied to inputs 4 through 7. When the device address is decoded, the multiplexer is enabled by DAXX- at the Strobe input. The outputs of the Function Bus unloading gates are applied to the decode input of the multiplexer (A, B and C). The appropriate sense signal, as determined by the function code, is then applied to the two outputs. Only the high output (Y) is used in this case. The signal is inverted and applied to the Sense Response line (SER-) by the 858 driver. When the Y signal is high, the SER- line goes low. When the Y signal is low, the SER- line stays high.

Example B shows positive sensing using SSI chips. Both external and internal sensing is again illustrated. A separate Function Decoder is required to provide the necessary function codes. NAND gates combine the sense lines with the associated function codes. The outputs of the NAND gates are connected in a wire-ORed configuration to the SER- line.

5.2.5.2 Negative Sensing

Example C shows negative sensing using the 74151 MSI chip. Negative sensing is similar to positive sensing, except that the low output (W) of the chip is employed, rather than the high output, the strobe input is grounded to permanently enable the chip and DAXX is used to gate the multiplexer output onto the SER- line. As with positive sensing, all external sense lines should be provided with pull-up resistors.

Example D shows negative sensing using SSI chips. The negative-true signals are inverted and applied to 858 2-input NAND gate drivers. Function code signals enable the appropriate driver. The outputs of the drivers may be connected in a wire-ORed configuration before being applied to the SER- line.

5.3 DATA TRANSFER CONTROL IMPLEMENTATION (Figure 5-6)

The efficient transfer of data between the Processor and device interface controller is controlled by the various buffer control circuits shown in Figure 5-6. An Output Buffer Empty circuit controls the transfer of data from the Processor to the interface (Examples A and B). An Input Buffer Full circuit controls the transfer of data from the interface to the Processor (Examples C and D).

5.3.1 Example A

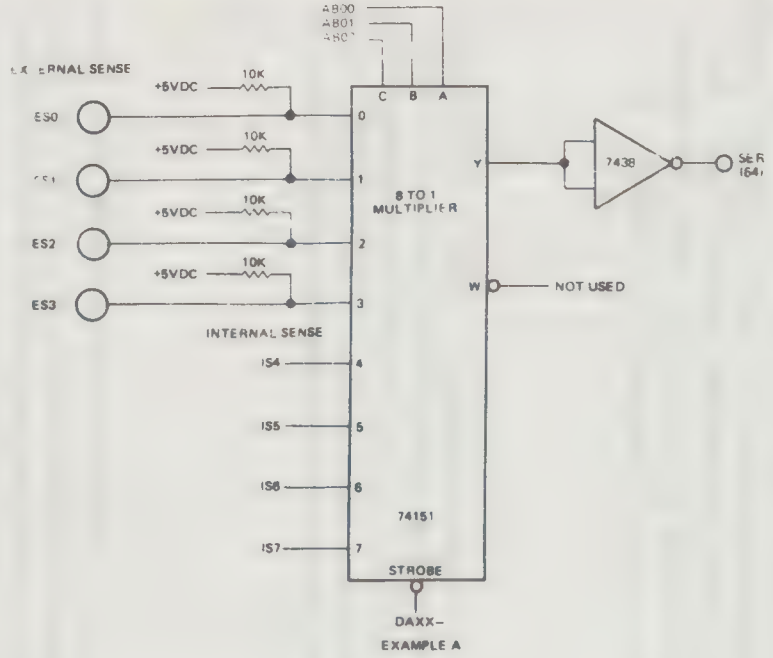
Example A shows an Output Buffer Empty latch (OBE) comprised of two DTL 846 negative input NOR gates. The latch is initially set upon execution of the Initialize command for the controller. The INZX signal goes high and is applied through the NOR gate to the set side of the latch, causing it to set. The OBE signal thus goes high and is applied to the Sense Multiplexer from which it can be interrogated by Sense or Conditional Output commands using the appropriate function code. The OBE signal can also cause an interrupt through implementation of interrupt logic. When data is transferred to the controller Output Buffer, the DAXX, OUT and PLSE signals go high, enabling the NAND gate whose output is applied to the reset side of the latch. The latch now resets, inhibiting response to further interrogations by the Processor. When the data has been transmitted, a signal should be generated to indicate completion of the transfer. (Data Transmitted--DXMT). DXMT is applied to the same NOR gate as INZX, causing the latch to set again and indicate that the buffer is ready for more data at the next Processor interrogation.

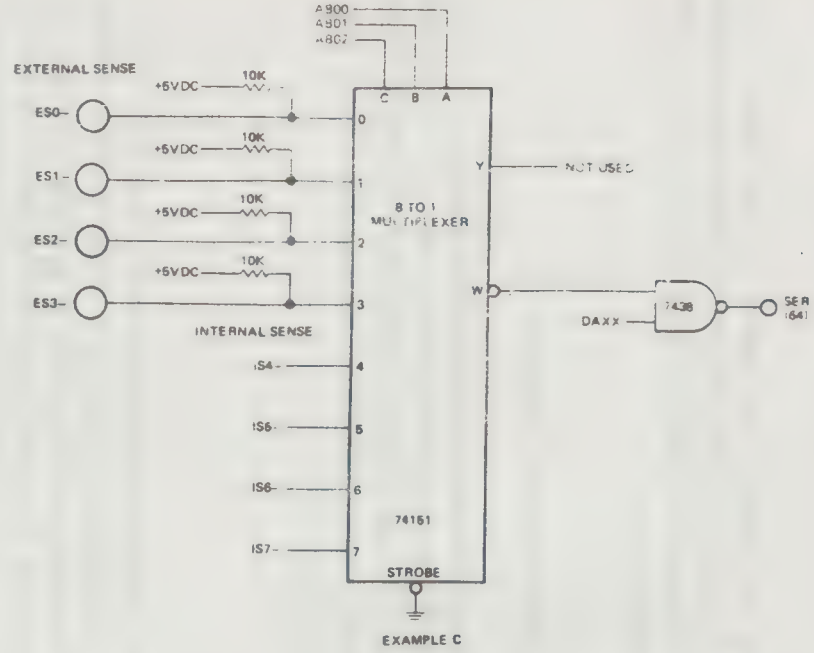
5.3.2 Example B

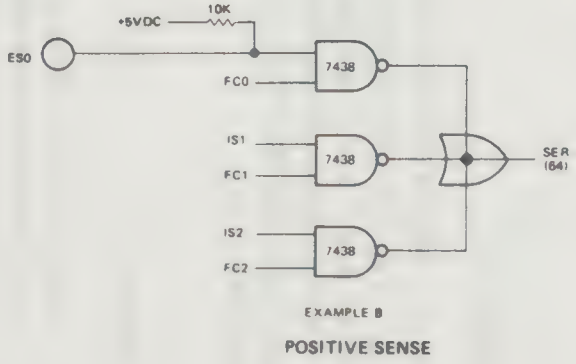
The circuit in example B does the same thing as example A. The only difference is a TTL 7474 D type flip-flop is used, rather than the dual NOR gate latch. INZX- direct sets the flip-flop. The high OBE signal is then available for interrogation. When data is transferred to the Output Buffer, the flip-flop is direct reset. When DXMT- goes true, the flip-flop is once again set to indicate the buffer is ready to accept more data.

5.3.3 Example C

Example C shows a latch configuration of an Input Buffer Full circuit (IBF). The latch is [illegible] by INZX upon initialization of the controller. After data has been [illegible]red to the Input Buffer, a signal should be generated to indicate the com-[illegible]tion of the transfer (Data Received--DRCV). DRCV- sets the latch, causing







NOTE FOR EXAMPLES B & D, ALL FUNCTION CODE DECODES MUST BE DEVELOPED USING DAXX AS IN FIG. 5-2 EXAMPLES A & B

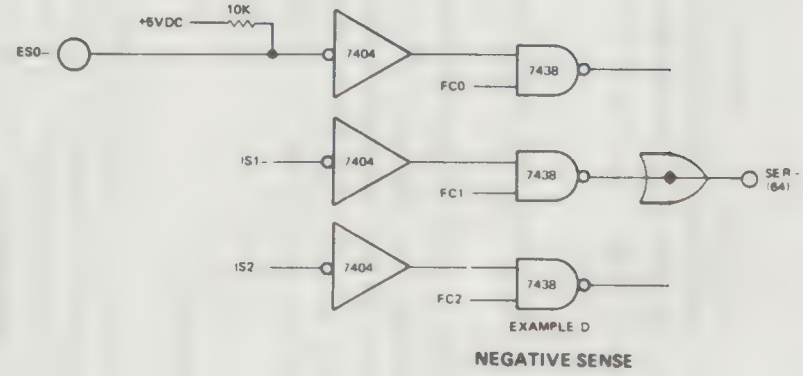


Figure 5-5. Positive and Negative Sense, Circuit Configurations

IBF to go high. The IBF signal is then applied to the Sense Multiplexer where it can be interrogated by the Processor with Sense or Conditioned Input commands. IBF can also cause an interrupt when implemented in the interrupt logic. When the data is transferred to the Processor, the DAXX, IN and PLSE signals go high, resetting the latch.

5.3.4 Example D

Example D shows an Input Buffer Full circuit using a TTL 7474 D type flip-flop. The flip-flop is direct reset upon initialization. The flip-flop is set when data is received (DRCV goes high). The flip-flop is then direct reset when the data is transferred to the Processor (DAXX, IN and PLSE go true).

5.4 PERIPHERAL DEVICE INTERRUPT IMPLEMENTATION

The design requirements for various interrupt structures compatible with the ALPHA LSI computer are now discussed.

5.4.1 Interrupt Address Rationale

In general, interrupts are vectored to the first 256 words of memory. The main advantages for having interrupts vectored to this area of memory are in the housekeeping associated with certain interrupt instructions. An Auto I/O instruction, for instance, must have the word/byte count and address pointer redefined after a block of data has been moved. An IMS instruction must have the count value redefined after it has overflowed or underflowed. If the interrupt instructions are in the first 256 words of memory, direct addressing can be used from anywhere in memory to update the instruction parameters in anticipation of the next interrupt pass.

In applications where the use of the first 256 words of memory for interrupts makes programming difficult, all interrupts can be offset : 100 locations into the next 256 words of memory.

The number of memory locations that are reserved for interrupts varies with each interface controller. If the interface controller is intended to move data under Auto I/O interrupt control, four locations should be reserved for the Auto I/O instruction and two locations for the Echo interrupt. If a simple transfer of control is required, only two locations are required for a JST instruction. If external events are being counted, four locations must be reserved--two for the IMS instruction and two for the Echo interrupt.

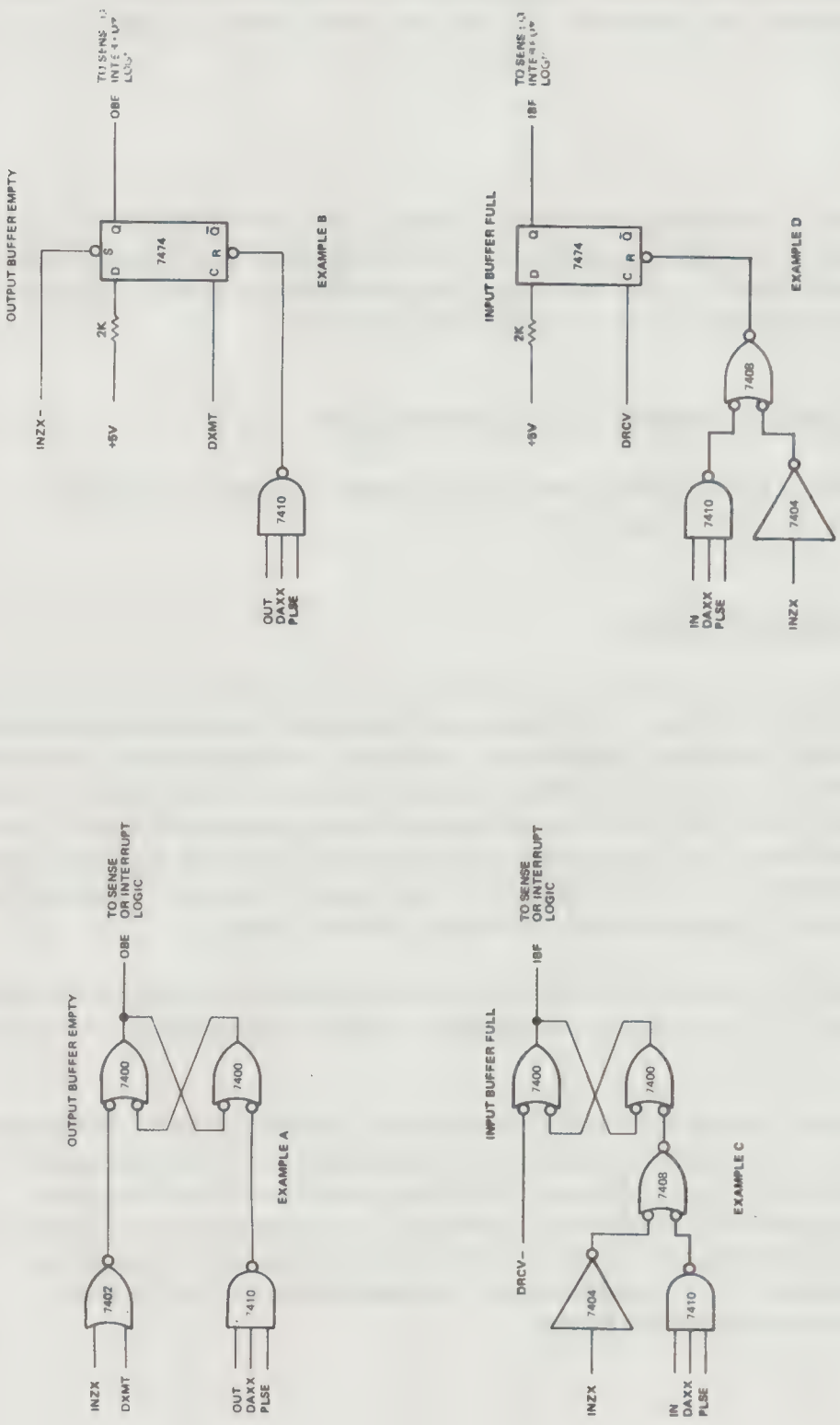


Figure 5-6. Data Transfer Control

If multiple interrupts are developed by an interface, these interrupts are organized into a family. Referring to Appendix A,tthe Real Time Clock option has a four word interrupt family and the 103 Data Set Controller has a 16-word family. Family size is strictly a function of the number of interrupts an interface develops and the number of locations required by each interrupt instruction.

To preserve compatibility throughout the ALPHA computer family, interface controllers are designed to interrupt to an even numbered address. If an interface controller develops multiple interrupts, the base addresses of these interrupts are partitioned either two or four locations apart. The standard base addresses are : 0XX2, : 0XX6, : 0XXA and : 0XXE. These standard base addresses leave locations : 0XX0 and : 0XX8 available for special interrupts, if required.

The Auto I/O instruction requires three locations while the IMS and JST instructions require one location each. The unused reserved locations may be used for address pointers.

5.4.2 Single Interrupt Implementation Using IUR- (Figure 5-7)

This structure features an Interrupt Enable flip-flop (INTE), an Interrupt Stimulus Store flip-flop (INTS), an Interrupt Pending flip-flop (IP1), priority determination logic, priority out disable logic and an interrupt address generator.

The INTE flip-flop is a J-K type device which is synchronously set or reset by an addressed select command. Function code M (FCM) sets INTE while function code R (FCR) resets INTE. The INTS flip-flop is a D-type positive-edge triggered circuit. When enabled, INTS sets on the positive excursion of the external stimulus signal (EXTS).

An optional feature is an edge detector consisting of an Exclusive-OR gate and an inverter. The edge detector permits the use of either a high or low stimulus signal. The polarity of EXTS is defined by RPOL (Request Polarity). If EXTS is a low signal when active, RPOL is grounded. Likewise, if EXTS is a high signal when active, RPOL is left open and the pull-up resistor provides the positive-logic level signal. When both EXTS and RPOL are of the same polarity, the output of the edge detector will be high causing INTS to set, if enabled. Once both INTE and INTS are set, an interrupt request is generated.

The Interrupt Pending flip-flop is enabled when INTE and INTS are both set. When enabled, IP1 sets on the negative excursion of the Processor I/O clock (IOCL). IOCL is low during the first half of a Processor phase and goes high during the second half of a phase. In this manner, IOCL synchronizes the interrupt structure to the Processor phase counter.

Once IP1 is set, the structure must have priority before an IUR interrupt request can be generated. If up-stream devices are not generating interrupts, PRIN- (priority in, pin 83) will be low. Both PRIN and IP1 are ANDed to produce the interrupt request signal, ME. ME is used to develop the IUR- signal and disable down-stream interrupts by causing PROT- (priority out, pin 84) to go high.

When the Processor recognizes the interrupt request, it responds by issuing the interrupt address request (IAR). If ME is still high (a higher priority interrupt may have been generated at the same time as this one, causing PRIN- to go high, disabling ME), IAR causes the interrupt address to be generated.

The interrupt address generator develops a unique vectored interrupt address. The base address that is developed is : 0XX2. The Interrupt Address Select lines (E4- through E256-) permit the user to displace the base address anywhere in the first 512 words of memory. Grounding a particular address select line adds a corresponding decimal value to all base addresses. For example, grounding E32- adds 32 decimal locations to all interrupt addresses.

This type of address generation permits the user to re-define interrupt locations with a minimum of effort. In the event the user is limited by the number of pins available, specific data bus drivers can be used instead of the structure shown.

When ME and IAR are high (ADDR), the data bus drivers are enabled and the interrupt address is transferred to the Processor. The Processor directs the contents of the data bus to the memory address register. After the memory address register is loaded, the PLSE signal is generated. The PLSE signal, NANDed with ADDR, will cause INTS to reset.

At the end of the last cycle of the interrupt instruction, IOCL is re-enabled. With INTS reset and IOCL enabled, IP1 resets on the negative excursion of IOCL terminating the IUR interrupt request.

The only feature of the interrupt structure not mentioned previously is the initialize feature. Generally all interface controllers have an initialize circuit which generates the INZ signal. INZ sets or resets all control flip-flops to a known condition. In this case, INTE and INTS are reset by INZ. INZ is typically generated in response to an addressed Select command with function code 4 or by the Processor generated System Reset signal, RST-.

5.4.3 Echo Interrupt Implementation Using IUR (Figure 5-8)

The interrupt structure shown in figure 5-8 develops two interrupts on the IUR- request line.

The structure is similar to the IUR structure described in paragraph 5.4.2 except than an Echo Interrupt flip-flop (ECHOI) is added. The interrupt request is developed as a result of ORing IP1 and ECHOI, and two base addresses are developed (: 0XX2 for IP1 and : 0XX6 for ECHOI).

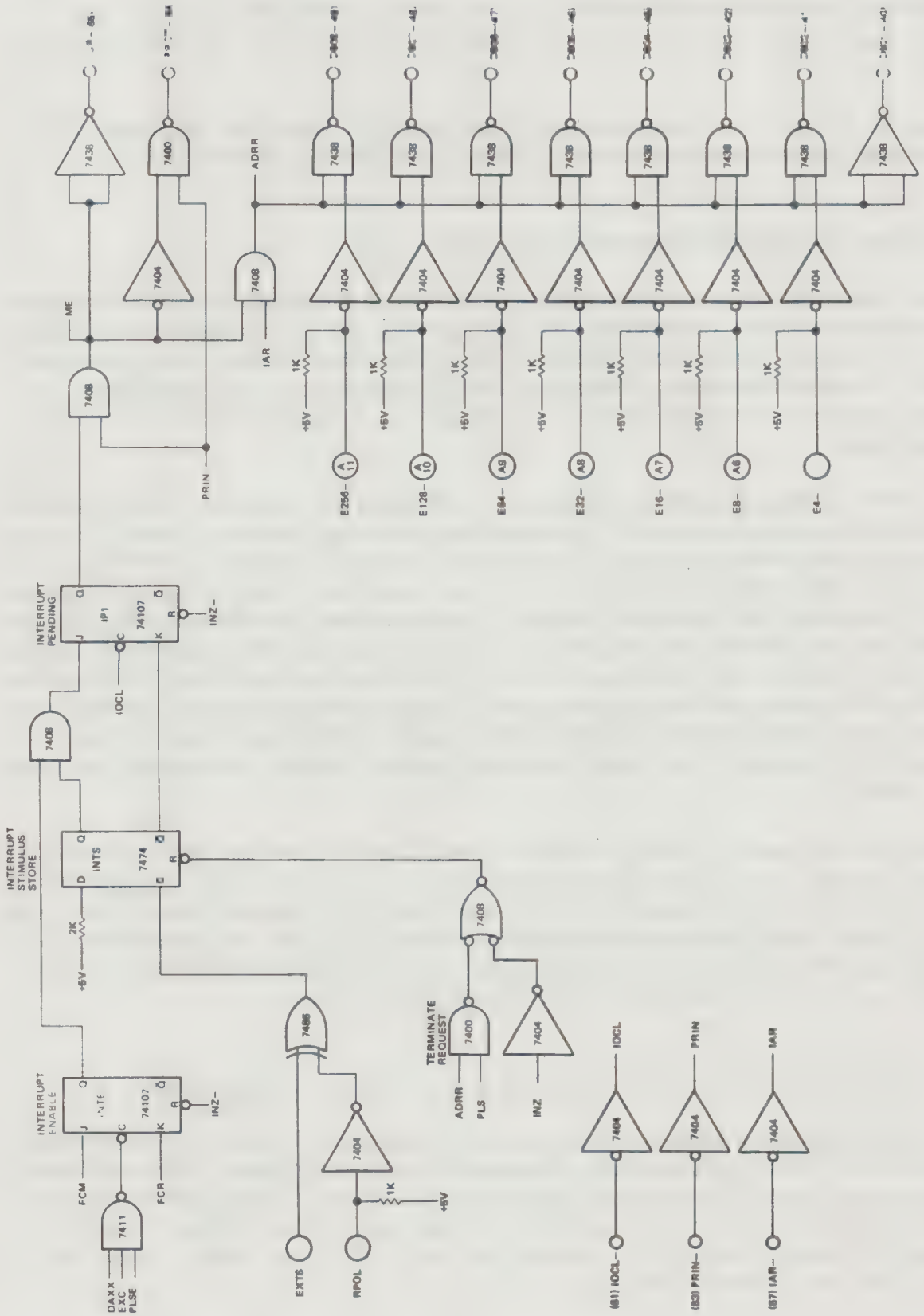


Figure 5-7. Single Interrupt Implementation Using IUR-

ECHOI is enabled by IP1 and PRIN. If the structure has priority at the instant an ECHO is developed by the interrupt instruction, ECHOI sets when the ECHO is received. ECHOI is reset, if IP1 is reset, if the structure has priority when IAR and PLSE are received.

Note that IP1 is set for the entire period of the interrupt instruction and that ECHOI is set only as long as required to obtain recognition from the Processor.

5.4.4 Reentrant Interrupt Implementation (Figure 5-9)

Reentrant interrupt programming permits an interrupt of higher priority to interrupt an interrupt subroutine. Interrupts of lower priority are not recognized. Reentrant interrupt programming requires that the Priority Out Disable latch be implemented in the user interface hardware. When the latch is implemented, the generation of an interrupt sets the latch which in turn, disables the generation of PROT- to downstream devices.

The reentrant interrupt feature disables all lower priority interrupts for the duration of an entire interrupt subroutine. The reentrant interrupt circuit is shown in figure 5-9. The circuit prevents the PROT signal from being transmitted to the next lower priority controller until the subroutine has been completed. The PROT disable latch is initially set when the interrupt request is acknowledged with the Interrupt Address Request (IAR) signal from the Processor. IAR is ANDed with ME to produce Address (ADRR) which enables the interrupt address drivers and also sets the PROT Disable latch. PROTD- thus goes low, disabling the 3-input NAND gate which normally produces the PROT- signal when ME- goes false (high). Inhibiting the generation of PROT- prevents priority from being passed on to lower priority controllers until the latch is reset.

The latch can be reset by issuing a Select command with a function code dedicated to resetting the latch, or by initializing the controller. When the Select command is decoded, the DEXP (combination of DAXX, EXEC and PLSE signals) signal goes high. DEXP is NANDed with the appropriate function code (FCX) and is applied through a negative input OR gate to the reset side of the latch. The latch is thus reset and PROT- is passed on to lower priority devices (if PRIN- is low).

5.4.5 Single Interrupt Implementation Using IL1- or IL2- (Figure 5-10)

The structure shown in figure 5-10 consists of an Interrupt Enable and an interrupt request driver. The interrupt enable is used to enable the driver. When the external stimulus is applied, an interrupt request is generated. This structure demands that the external stimulus remain active until some positive action takes place to move data or transfer control (the issuance of the IN-, OUT- or EXEC- control signals with the proper device address).

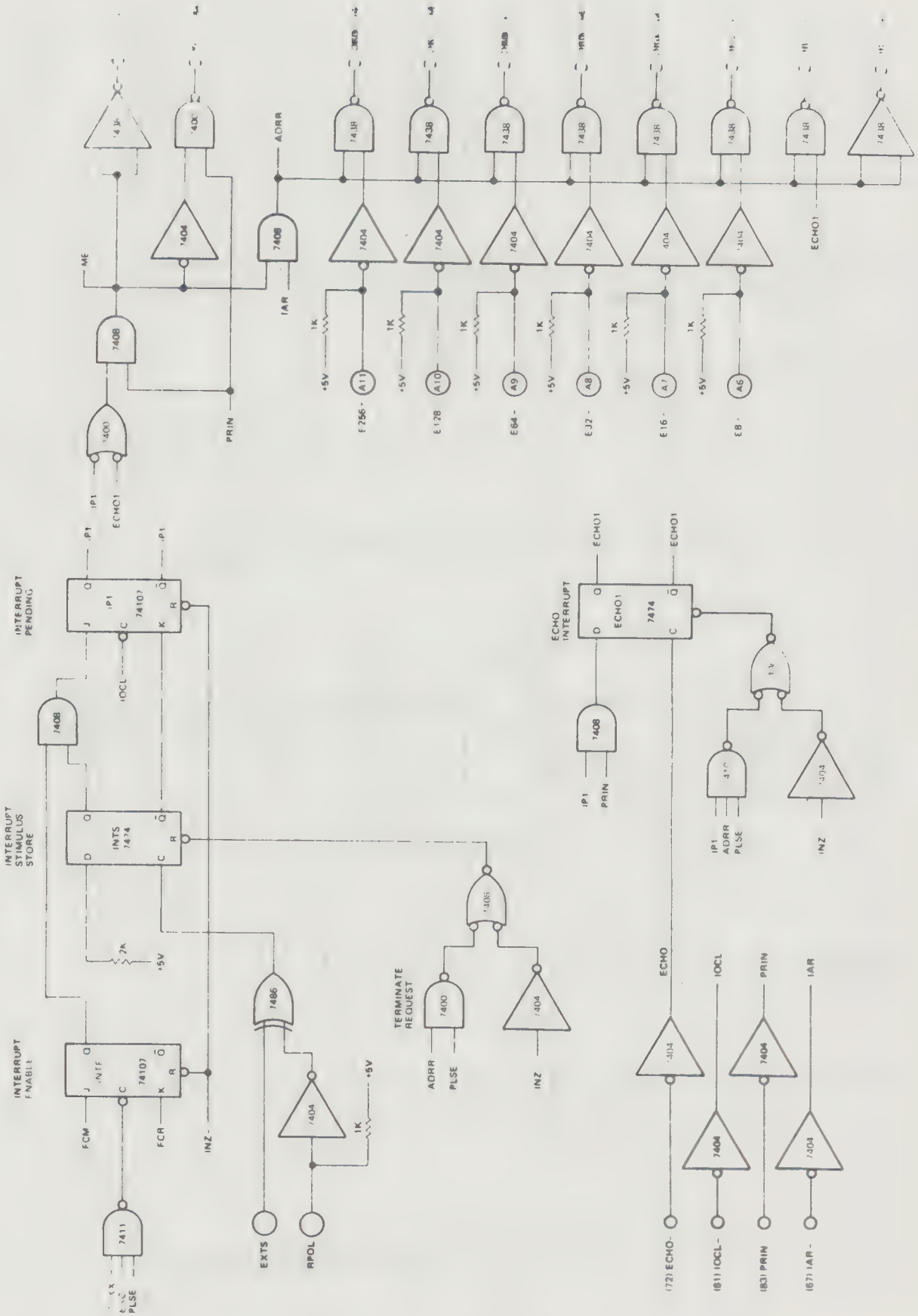


Figure 5-8. IUR- Echo Interrupt Implementation

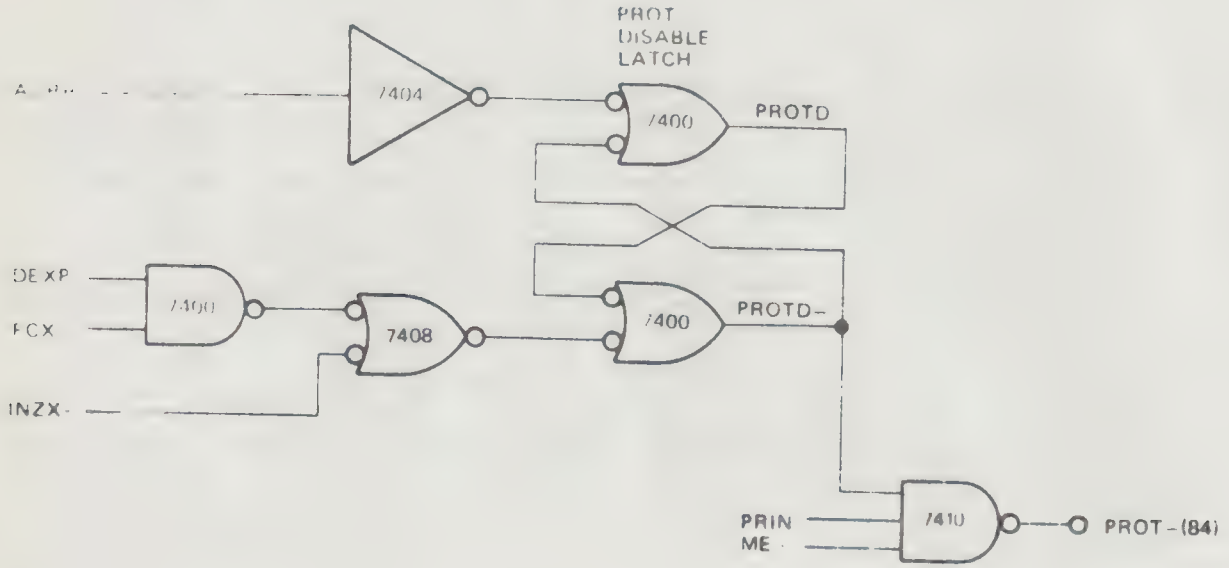


Figure 5-9. Reentrant Interrupt Implementation

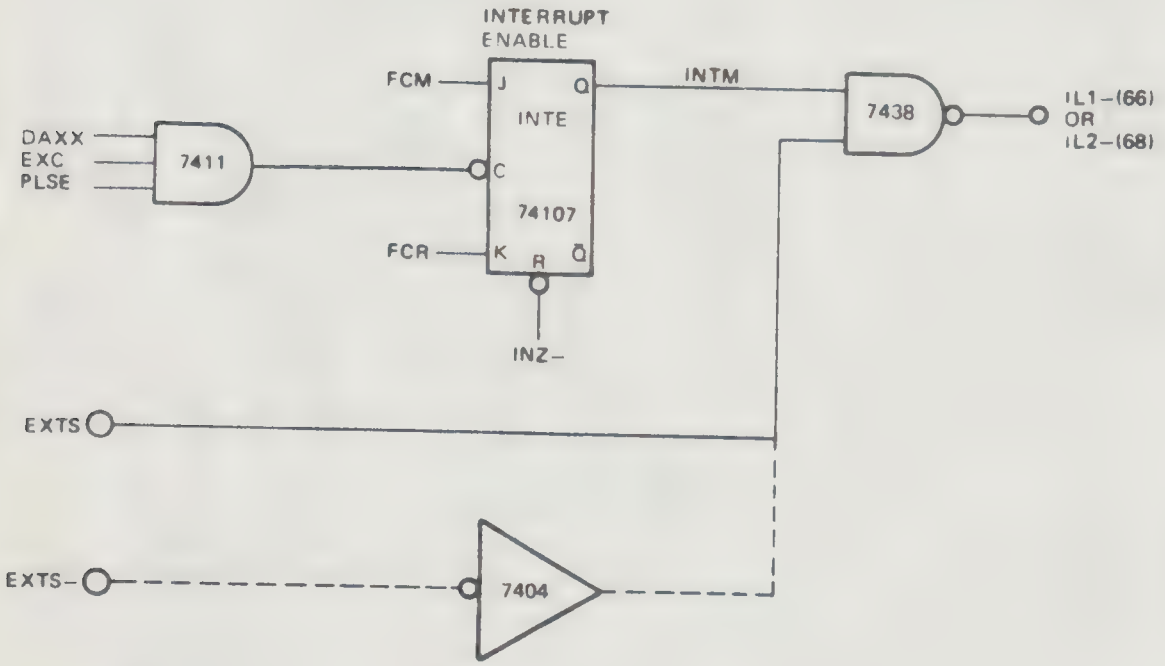


Figure 5-10. Simple IL1-/IL2- Interrupt Structure

5.4.6 Echo Interrupt Implementation Using IL1 and IL2 (Figure 5-11)

The interrupt structure shown in figure 5-11 develops two interrupts which utilize the IL1- and IL2- request lines. Since this interrupt structure is designed to accommodate any echo generating instruction (the four Auto I/O instructions and the IMS instruction), no other devices may be attached to the IL1- and IL2- request lines. These lines are totally dedicated to this structure.

This structure is essentially the same as the IUR- structure described in paragraphs 5.4.2 and 5.4.3. The most significant difference is that the request flip-flops are distributed directly to the IL1- and IL2- drivers. The operation of this structure is essentially the same as the IUR structures, except during request termination. Once the interrupt request is generated, the request must be recognized by the Processor. The Processor recognizes the highest priority interrupt first and all other requests in their order of priority. Since there are four higher priority interrupts above IL1 (power fail, trap, console interrupt and memory parity) and five above IL2- (the four just mentioned and IL1), the interrupt structure must be able to detect no higher priority interrupt activity before terminating the request. The only thing that the power fail, trap, console interrupt and memory parity interrupts have in common is that during the interrupt address request interval, they all cause bit 4 of the Data bus to be low. If DB04- is low during IAR, the IL1 request will not reset but will remain active since the Processor has not honored the request. When no higher priority exists after generating the interrupt request, INTS is reset on the leading edge of the PLSE signal and terminates the interrupt request. To avoid re-triggering the INTS flip-flop, the interrupt stimulus should remain in the active condition until an addressed I/O command (Select, Input or Output) causes the source of the stimulus to reset.

5.5 PRIORITY PROPAGATION

It is the users' responsibility to propagate interrupt priority, regardless of whether or not an interface controller develops interrupts. If an interface controller does not develop interrupts, the PRIN- and PROT- signals must be jumpered together inside the interface controller.

Interface controllers that develop IUR interrupts should use TTL gates for unloading PRIN- and driving PROT-. It is imperative that the propagation delays internal to the interface controller be minimized. A total of two microseconds is allowed for priority propagation through all controllers in a chain. The implementation of expansion chassis Buffer Card look-ahead propagation limits the longest priority propagation path to the maximum number of controllers that can be installed in two chassis (20 controllers). Priority propagation delays should therefore be held to less than 100 nanoseconds average per controller.

5.6 I/O BUS LOADING RULES

In order to conserve the I/O bus drive capability of the Processor, an interface should not present more than one load (1.6 ma typical) to any given I/O bus signal. If the interface controller presents more than one load to a given signal, the user should unload the signal upon entry into the interface.

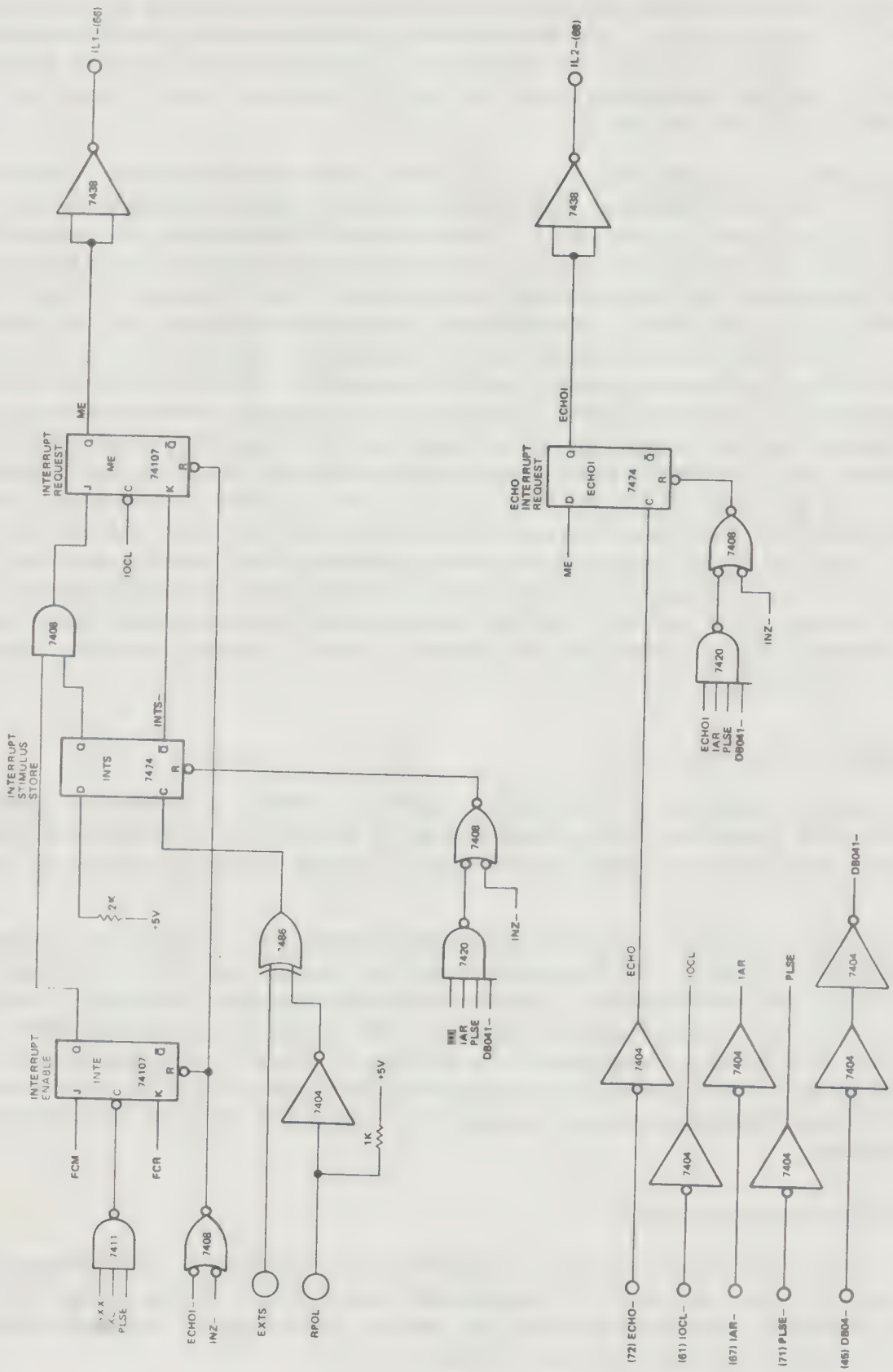


Figure 5-11. Echo Interrupt Implementation Using IL1- and I12-

5.7 POWER AND GROUND SYSTEM CONCEPTS

The power supply that is furnished with the ALPHA 16 LSI computer produces three voltages: +5Vdc, +12 Vdc and -12 Vdc. The +5 volt supply is used to provide the VCC voltage for all integrated circuits in the Processor, Memory and I/O modules. The +12 and -12 volt supplies are used by the Processor and Memory modules and are available to all I/O modules if needed. Typically the +12 and -12 volt supplies provide power for analog and communications type interfaces. All three regulated voltages share a common ground system referred to as logic ground.

Power (+5, +12 and -12 Vdc) and logic ground are distributed from the system power module through the motherboard to all plug-in modules. Within a module, +5V and ground are distributed by means of bus bars. The power and ground pins on the motherboard are organized such that each bus bar can pick up a separate set of pins.

A typical half card module has a density of 72 integrated circuits which are organized in six rows of 12 chips. A typical full card module has a density of 144 IC's organized in 12 rows of 12 chips. Bus bars are mounted in between each row of chips and on the outside edges of a card. A half card module has seven bus bars while a full card module has 13. Odd numbered bus bars are ground, even numbered bus bars are +5 Vdc.

Most 14-pin chips use pin 14 for Vcc (+5 Vdc in this case) and pin 7 for logic ground. A typical 16-pin chip uses pin 16 for Vcc and pin 8 for logic ground. By alternating the pin 1 orientation of each row of chips, two rows of chips can share a common +5 or ground bus bar. All bus bars are approximately 14 inches long and have 24 pins located on .6 inch centers. The Vcc pins of all chips in adjacent rows are routed to the nearest +5 bus bar mounting pad. Likewise, all ground pins in adjacent rows are routed to the nearest ground bus bar mounting pad.

The bus bar is designed such that when it is installed there is a .075 inch gap between the underside of the bus bar and the printed circuit board. This is to permit etched circuitry to pass underneath the bus bar without shorting.

Table 5-1 lists all power and ground pin assignments that exist in the 100 and 200 connectors of a typical motherboard slot.

Table 5-1. Power and Ground Pin Assignments

PIN	SIGNAL	PIN	SIGNAL
1,2	Ground	43,44	+5 Vdc
3,4,5,6	+12 Vdc	59,60	Ground
7,8	-12 Vdc	73,74	+5 Vdc
13,14	+5 Vdc	85,86	Ground
27,28	Ground		

There are two ground systems in the ALPHA 16 LSI computer. They are logic ground and chassis ground. It is recommended that the user avoid tying these two ground systems together. The chassis ground system usually has more noise than the logic ground system can tolerate. In the event it is necessary to tie the two systems together, they should be tied together at only one point in the users' system. For personnel protection, the chassis ground system is tied to earth-ground via the third wire in the AC line cord.

5.8 FILTERING TECHNIQUES

Integrated circuits introduce switching transients into the +5 Vdc power supply which must be filtered out. It is recommended that both high frequency and low frequency filtering be employed. The low frequency filter consists of a 2.2 microfarad, 10 per cent, 20 Vdc tantalum capacitor between +5V and ground for each row of 12 chips. The high frequency filter consists of a .022 microfarad, 25 Vdc ceramic capacitor between +5V and ground for every four chips in a given row of chips. Thus a typical half card module would have 6 tantalum capacitors and 18 ceramic capacitors for transient filtering. Where a large number of MSI devices and Fairchild 9202 one-shots are used, it is recommended that a .022 microfarad ceramic capacitor be used for each device.

The -12 Vdc supply is used by the inhibit drivers in memory. The inhibit drivers introduce approximately .5 volts of transient noise into the -12 Vdc power supply. If the user cannot tolerate this much noise, an inductive input type filter is recommended.

5.9 STANDARD INTERFACE CONNECTOR

The standard interface connector is a Viking 3VH50/1JN5 or equivalent. This connector features two rows of 50 contacts designated A1 through A50 and B1 through B50. Contacts A1 through A50 interface with the contact strip on the solder side of the printed circuit board. Contacts B1 through B50 interface with the component side of the board. The interface connector should be installed with pins B1 and A1 to the left as viewed from the rear of the computer.

5.10 NORMAL INTERFACE PINS

The interface pin assignments normally used by CAI for device address and interrupt address jumpers are listed in Table 5-2.

Table 5-2. Normal Interface Pins

PIN	SIGNAL	PIN	SIGNAL
A01	PS4-	B01	+5Vdc
A02	PS3-	B02	+5Vdc
A03	PS2-	B03	Ground
A04	PS1-	B04	Ground
A05	PS0-	B05	Ground
A06	E8-	B06	Ground
A07	E16-	B07	Ground
A08	E32-	B08	Ground
A09	E64-	B09	Ground
A10	E128-	B10	Ground
A11	E256-	B11	Ground

Section 6

INTERFACE CONTROLLER MECHANICAL CONSIDERATIONS

6.1 INTRODUCTION

This section discusses the mechanical design of a printed circuit board which can be installed in an ALPHA 16 LSI computer chassis.

Either full or half printed circuit boards may be used. When half boards are used, two half boards are joined together to form a full board.

All boards use bus bars to distribute power and ground to circuits. The bus bars minimize the ground and power etch runs, leaving more space on the board for signal etched circuit routing. The bus bar design permits etched circuitry to be routed underneath the bus bar with no danger of shorting.

Fiberglass stiffeners are used on all boards to eliminate sag and provide improved structural integrity.

6.2 CHASSIS CONSTRAINTS

The computer chassis is designed to accommodate a printed circuit board which has a width of 15 inches. All printed circuit boards are installed in the horizontal position. When installed, the chassis provides four-way support for the printed circuit board. The card guides support both sides of the printed circuit, the motherboard board connectors support the front, and a notched card retainer supports the rear edge.

The thickness of the printed circuit board is determined by the motherboard connectors. A typical board is .062 inches thick. The motherboard connector permits variations in thickness ranging from .054 to .071 inches.

All components, stiffeners, bus bars, etc. are mounted on one side of the board. This side of a board is referred to as the "component side" while the other side is referred to as the "solder side". Boards are always installed with the component side up.

The chassis card guides are spaced on .75 inch centers. The height of components on the component side of a board and the lead protrusion on the solder side of a board must be minimized to permit unimpeded airflow and easier insertion and removal of printed circuit boards. All components should be no higher than .47 inch maximum. Lead protrusion should be held to .062 inch. maximum.

The card guides are an integral part of the computer chassis which is metal. To prevent short circuits on a board, the user should not permit any etched circuit runs that are closer than .200 inch from either edge of a board.

6.3 PRINTED CIRCUIT BOARD CONSIDERATIONS (Figures 6-1 thru 6-3)

Figures 6-1 and 6-2 show the critical dimensions, hole patterns for bus bars and stiffeners and the integrated circuit layout organization for a full and a half board, respectively.

The motherboard interface dimensions are extremely critical and must be adhered to rigorously.

The rear edge of the full board has room for two interface connectors. The 1.250 inch dimension from each edge is the area reserved for the card extractors (Part No. 40-06100-00). The .800 inch dimension at the center is the area reserved for the card retainer. The remaining area along the rear edge is connector area. The 6.350 inch dimension is the maximum allowable area that the mating connector can occupy. The overall length of a connector cannot exceed this dimension.

The rear edge of a half board has room for only one interface connector. A distance of 1.210 inch must be reserved for a modified card extractor (Part No. 00-00296-00). This leaves 5.080 inches of useable connector area remaining. The 5.080 dimension is the inside contact dimension of the standard 100-pin interface connector.

Half boards must provide for a card extractor at both rear corners although only one is installed depending upon which way the board is strapped to a second half card.

Figure 6-3 shows the standard printed circuit board hardware. All dimensions are provided for layout planning purposes. Connector data on the motherboard connector and various rear-edge interface connectors is also provided.

6.4 WIRE-WRAP BREADBOARD CARD

A half board wire-wrap breadboard card (Part Number 13234-00) is available from Computer Automation, Inc. This card features 72 IC sockets with wire-wrap posts, ground and power busses, and filters. This card can be useful for prototype development and checkout prior to making a formal printed circuit board design.

6.5 CARD JOINING PROCEDURES

Card joining information is described in the ALPHA LSI Installation Procedure.

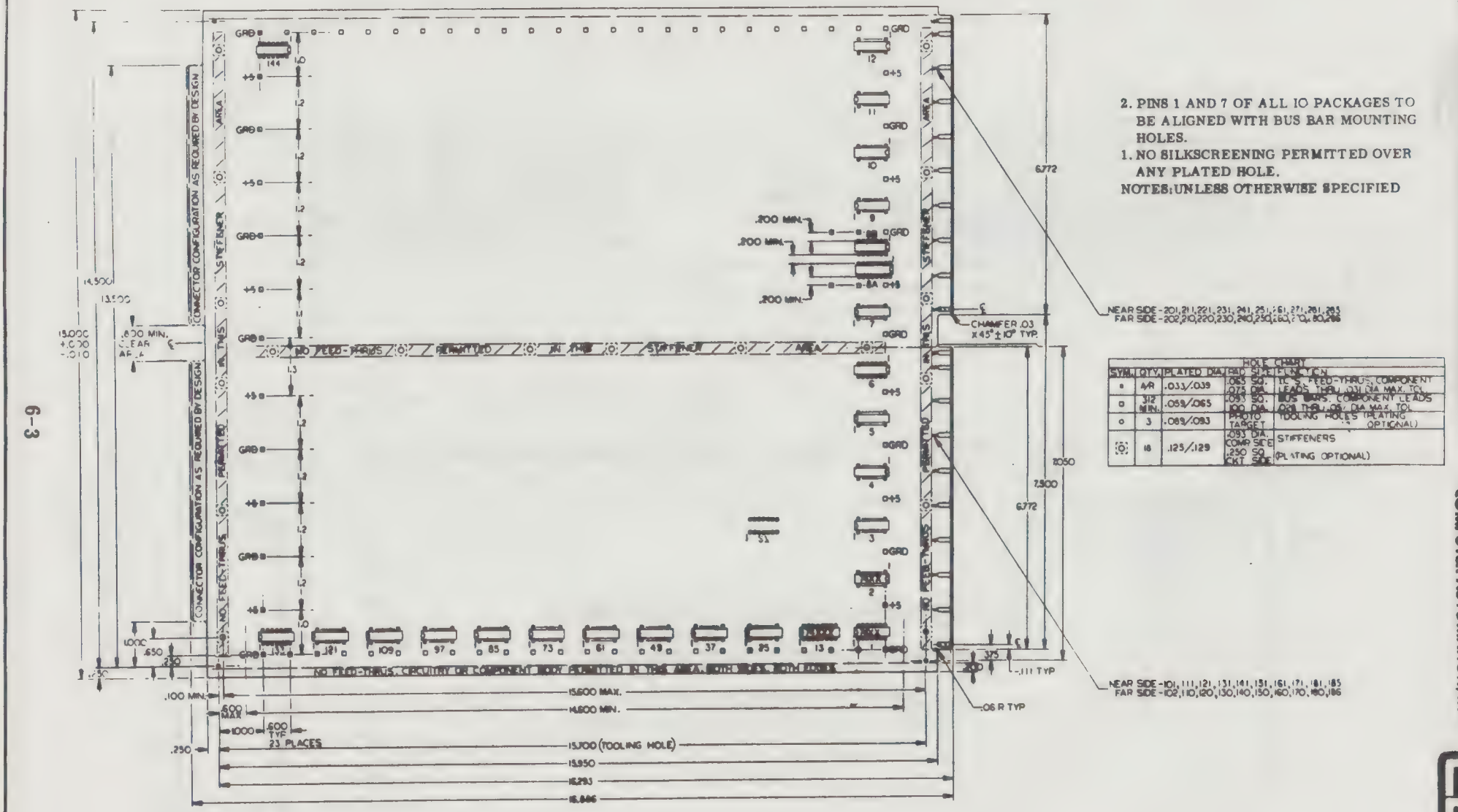
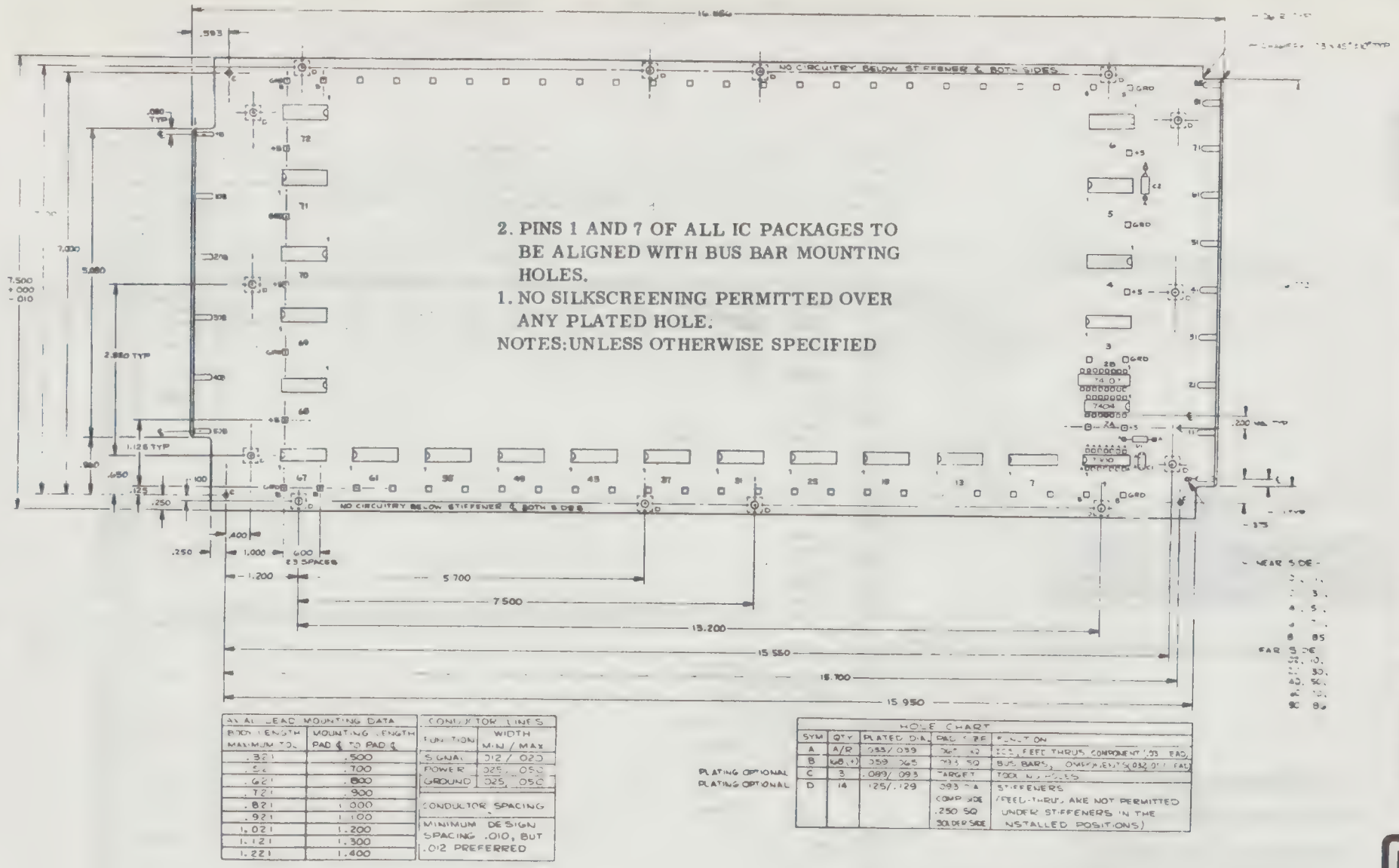


Figure 6-1. Full Board Design Guide



AXIAL LEAD MOUNTING DATA	
BODY LENGTH MAXIMUM TO.	MOUNTING LENGTH PAD ℄ TO PAD ℄
.361	.500
.561	.700
.661	.800
.721	.900
.821	1.000
.921	1.100
1.021	1.200
1.121	1.300
1.221	1.400

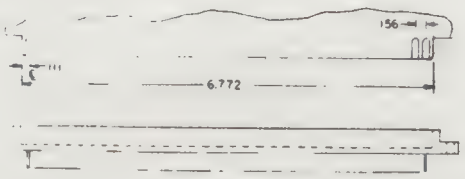
CONDUCTOR LINES	
FUNCTION	WIDTH MIN / MAX
SIGNAL	.012 / .020
POWER	.025 / .050
GROUND	.025 / .050
CONDUCTOR SPACING	
MINIMUM DESIGN SPACING .010, BUT .012 PREFERRED	

PLATING OPTIONAL
PLATING OPTIONAL

HOLE CHART				
SYM	QTY	PLATED DIA	PAD DIA	FUNCTION
A	A/R	.033/.039	.062 SQ	ECS FEED THRUS COMPONENT .093 PAD
B	68 (+)	.359/.365	.093 SQ	BUS BARS, OVERSIZED (.032/.017 PAD)
C	3	.089/.093	TARGET	TOOL LOCATING HOLES
D	14	.125/.129	.093 DIA COMP SIDE .250 SQ SOLDER SIDE	STIFFENERS (FEED-THRUS ARE NOT PERMITTED UNDER STIFFENERS IN THE INSTALLED POSITIONS)

Figure 6-2. Half Board Design Guide

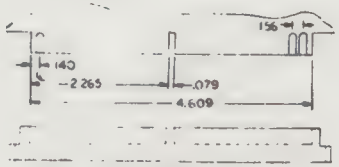
MOTHERBOARD CONNECTOR



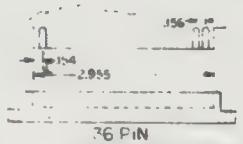
86 PIN
PN 17-10001-43P0

REAR EDGE CONNECTOR VARIATIONS

100 PIN
PN 17-10001-50P0

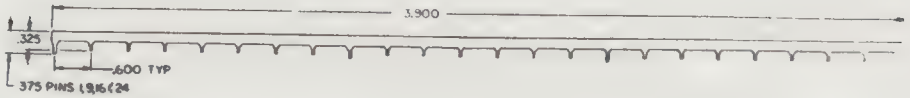


56 PIN
PN 17-10002-28P1

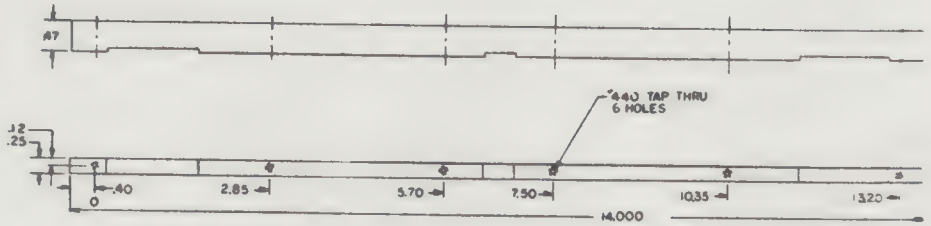


36 PIN
PN 17-10002-18P1

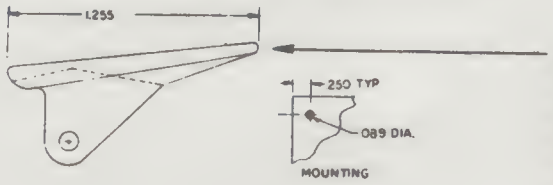
BUS BAR (PN00-00285-00)



STIFFENER (PN00-00160-00)



FULL BOARD CARD EXTRACTOR (PN40-06100-00N0)



NOTE: The half-board card extractor (PN00-00296-00) is the same as the full-board extractor except .130 inches of material are removed from the tip of the extractor.

Figure 6-3. Standard Circuit Board Hardware

Appendix A

RECOMMENDED DEVICE AND INTERRUPT ADDRESSES

A.1 GENERAL

Table A-1 and A-2 list recommended Device and Interrupt Addresses to prevent possible conflict during future expansion to other I/O modules.

Table A-1. Recommended Device Addresses

DEVICE	DEVICE ADDRESSES	
	STANDARD	ACTUAL
Power Fail Restart*	00	
Memory Protect*	00	
	01	
Dual TTY/CRT (TTY1/CRT1)	02	
Dual TTY/CRT (TTY0/CRT0)	03	
Line Printer (LP)	04	
Card Reader (CR)	05	
Paper Tape Punch(PTP)	06(17)	
Paper Tape Reader (PTR)	06	
Processor TTY* (TTY)	07	
Real Time Clock* (RTC)	08	
Magnetic Tape (Mag Tape)	09	
	0A	
	0B	
	0C	
	0D	
	0E	
Disc	0F	
Cassette	10	
	11	
16-Bit I/O (A/D System)	12	
	13	
Plotter	14	
	15	
32-Bit Relay In (RCIM)	16	
Punch Alternate	17	
16-Bit Input/Output (16-Bit I/O)	18	
64-Bit Input (64-Bit In)	19	
64-Bit Output (64-Bit Out)	1A	
Priority Interrupt Module (PIM)	1B	
32-Bit Relay Out (RCOM)	1C	
103 Data Set Controller (103 DSC)	1D	
	1E	
	1F	

*Processor mounted options. Device Addresses non-alterable.
( ) Indicates suggested alternate.

Table A-2. Scratchpad/Page 0, Recommended Interrupt Address Map

	00-1F	20-3F	40-5F	60-7F	80-9F	A0-BF	C0-DF	E0-FF
0	:00* POWER UP	:20* 64-BIT OUT			:80 PIM(0)			
1								
2	:02 TTY WORD	:22 MAG TAPE WORD	:42 LP WORD	:62 TTY0/CRT0 WORD	:82 PIM(1)	:A2 PLOTTER WORD		
3								
4					:84 PIM(2)			
5								
6	:06 TTY EOB	:26 MAG TAPE EOB	:46 LP EOB	:66 TTY0/CRT0 EOB	:86 PIM (3)	:A6 PLOTTER EOB		
7								
8					:88 PIM(4)			
9								
A	:0A M.H. DISC	:2A PTR/PTP WORD	:4A CR WORD	:6A TTY1/CRT1 WORD	:8A PIM(5)	:AA RCIM WORD		
B								
C					:8C PIM(6)			
D								
E		:2E PTR/PTP EOB	:4E CR EOB	:6E TTY1/CRT1 EOB	:8E PIM(7)	:AE RCIM EOB		
F								
0	:10* 64-BIT IN	:30** 103 DSC ANSWER	:50 CASSETTE ADDRESS		:90 PIM(8)			
1								
2	:12 MEMORY PARITY	:32** 103 DSC INPUT WORD	:52 CASSETTE WORD	:72 16-BIT I/O WORD	:92 PIM(9)			
3								
4	:14 MEMORY PROTECT				:94 PIM(10)			
5								
6		:36** 103 DSC IN EOB	:56 CASSETTE EOP	:76 16-BIT I/O EOB	:96 PIM(11)			
7								
8	:18 RTC CLOCK	:38** 103 DSC PAR ERR/FLT			:98 PIM(12)			
9								
A	:1A RTC SYNC	:3A** 103 DSC OUTPUT WORD	:5A RCOM WORD	:7A 16-BIT I/O WORD	:9A PIM(13)			
B								
C	:1C POWER DOWN				:9C PIM(14)			
D								
E	:1E CONS INT & TRAP	:3E** 103 DSC OUT EOB	:5E RCOM EOB	:7E 16-BIT I/O EOB	:9E PIM(15)			
F								

XX = Interface Generated Interrupt Address
* = Non-Alterable Address
** = Locations 30-3F are reserved for Autoload option, if used (103 DSC addresses must be relocated.)

EOB = End-of-Block
EOP = End-of-Operation